ESL-TR-88-04

DISPERSION CORRECTION FOR SPLIT-HOPKINSON PRESSURE BAR DATA

L.E. MALVERN, D.A. JENKINS, E. JEROME, J.C. GONG

ENGINEERING SCIENCES DEPT
UNIVERSITY OF FLORIDA
GAINESVILLE FL 32611

JULY 1988

FINAL REPORT

JANUARY 1987 - SEPTEMBER 1987

ENGINEERING & SERVICES LABORATORY
AIR FORCE ENGINEERING & SERVICES CENTER
TYNDALL AIR FORCE BASE, FLORIDA 32403

SECURITY CLASSIFICATION OF THIS PAGE

ADA201314

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release Dist: Unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) ESL-TR-88-04
6a. NAME OF PERFORMING ORGANIZATION Engineering Sciences Dept University of Florida 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION HQ Air Force Engineering and Services Center
6c. ADDRESS (City, State, and ZIP Code) Gainesville FL 32611	7b. ADDRESS (City, State, and ZIP Code) Tyndall AFB FL 32403-6001
8a. NAME OF FUNDING/SPONSORING ORGANIZATION HQ AF Engineering & Services Ctr 8b. OFFICE SYMBOL (If applicable) AFESC	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F08635-83-C-0136
8c. ADDRESS (City, State, and ZIP Code) Tyndall AFB FL 32403-6001	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO.: 62206F PROJECT NO.: 2673 TASK NO: WORK UNIT ACCESSION NO.: 0078

11. TITLE (Include Security Classification)

Dispersion Correction for Split-Hopkinson Pressure Bar Data

12. PERSONAL AUTHOR(S)

L.E. Malvern, D. A.Jenkins, E. Jerome, and J. C. Gong

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
	FROM 870115 TO 870930	July 1988	72

16. SUPPLEMENTARY NOTATION

Availability of this report is specified on reverse of front cover.

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Dispersion Correction, Wave Propagation
Hopkinson Bar, Kolsky Device,
Fast Fourier Transform

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Large diameter Split-Hopkinson Pressure Bar (SHPB) systems require a correction of the strain signals to account for wave dispersion. Using a fast fourier transform (FFT) algorithm a wave dispersion correction program was developed for the 3.0 inch diameter SHPB at the University of Florida and the 2.0 inch diameter SHPB at Tyndall AFB FL. Since the data are applicable to only these two systems the report is published as a user's manual for these two systems.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

PREFACE

This report was prepared by the Engineering Sciences Department, College of Engineering, University of Florida, Gainesville FL 32611, under Task Order 86-5 Contract No. F08635-83-C-0136, for the HQ Air Force Engineering and Services Center, Engineering Services Laboratory (AFESC/RDCS) Tyndall Air Force Base FL 32403-6001. Mr. J. R. Hayes, Jr. (AFESC/RDC*) was the Government technical program manager. The work was accomplished between 15 Jan 1987 and 30 September 1987.

This report summarizes the results of work to investigate the effects of wave dispersion in strain data collected using a large diameter split-Hopkinson pressure bar (SHPB). The project was directed solely at reducing data for a large diameter SHPB. Since there are only a very few of these devices, a user's manual was requested. For the convenience of the user, this report is being published in the form in which it was submitted.

The user's manual contains a short user's guide and a detailed description that covers all the aspects and running of a computer program that performs analyses of data obtained from a split-Hopkinson pressure bar experiment. The computer program was written specifically for an IBM PC-XT but is written in standard Fortran such that it may be used on any mainframe computer or PC with a Fortran compiler.

Treatment of data previcusly stored in the computer and establishing files of output data obtained from analysis of this stored data is the major task of this computer program. The procedure of initial raw data transfer to the computer and graphics output from the stored output files is described seperately for each system.

This report has been reviewed by the Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, incluading foreign nationals.

This report has been reviewed and is approved for publication.

John R. Hayes Jr.
J. R. HAYES, JR
Senior Scientist
Engineering Research Division

LAWRENCE D. HOKANSON, Colonel, USAF
Director, Engineering and
Services Laboratory

Robert J. Majka
R. J. MAJKA, Lt Col, USAF
Chief, Engineering Research
Division

(The reverse of this page is blank.)

TABLE OF CONTENTS

LIST OF FIGURES

1. USER'S GUIDE

The User's Guide describes the treatment of input data files that are already stored in the computer. The input files are in units of microstrain. The filenames are IDio.dat for the incident pulse, IDro.dat for the reflected pulse and IDto.dat for the transmitted pulse, where ID is the specimen identification consisting of one to three characters (numbers and/or letters).

Program SHPB runs interactively. The first thing that appears on the screen is the program title and the parameters for the system that was used last.

EXAMPLE:
```
THIS IS PROGRAM SHPB.  IT PERFORMS THE PHASE
CORRECTIONS FOR DISPERSION ON ANY WAVEFORM
AND DATA ANALYSIS.

     THE PROGRAM PARAMETERS ARE CURRENTLY SET
TO THE VALUES ENTERED FOR TYNDALL 2-INCH SHPB

THE PARAMETER DEFINITIONS AND CURRENT VALUES ARE:
1    Number of terms used in FFT: mm=55
2    Number of points to be analyzed in each pulse:
                              ni=1024
     The value of ni must not be greater than 1024.
3    Time step:           tstep=0.50000 microsec
4    Diameter of the bar:  dia=0.05080 meters
5    Distance between the gages on the incident bar and
     the specimen:         dzi=1.01600 meters
6    Distance between the gages on the transmitter bar
     and the specimen:     dzt=1.01600 meters
7    Longitudinal wave velocity in the bar:
                         velocb=4406.50 meters/sec
8    Density of the bar: ro=7830.00 kg/meters cubed

(*)  IS THIS THE SHBP SYSTEM YOU WISH TO USE?   Y or N
```

For the Y-N choice questions, if the user responds with anything besides Y, y, N or n, the question will be asked again.

If the user wishes to use a different system from the one displayed, answer must be " N " or " n ". A directory with menu choices of all the systems is then displayed. Once one has been chosen, the parameters for that system are displayed.

```
EXAMPLE: IS THIS THE SHBP SYSTEM YOU WISH TO USE? Y OR N
         N
          THIS IS A DIRECTORY OF THE AVAILABLE SYSTEMS:

          1   UNIVERSITY OF FLORIDA 3/4 INCH SHPB
          2   TYNDALL AIR FORCE BASE
          3   NEW SYSTEM OF YOUR CHOICE
          4   UNIVERSITY OF FLORIDA 3.0-INCH SHPB
          WHICH SYSTEM DO YOU WISH TO USE? 1,2,3 OR 4?
          4
```

Parameters for Systems 1, 2 and 4 are stored in files UF2.dat, TYND.dat and UF.dat, respectively. The user may store parameters in NEWSYS.DAT for a system of his choice.

The parameters for system 4 would then be displayed and question (*) asked again. If the answer is " N " or " n " again, the system directory will appear. When the answer is " Y " or " y ", the user now has the option to change any of the parameter values. A parameter directory with menu choices is displayed, if any changes need to be made.

```
EXAMPLE:  PARAMETER DIRECTORY
          1   Number of terms used in Fourier transform
               Current value is        55
          2   Number of points to be analyzed.
               It MUST be no more than 1024 points.
               Current value is      1024
          3   Time step in microseconds
               Current value is  0.50000
          4   Diameter of the bar in meters
               Current value is  0.07620
          5   Distance between incident bar gage and the
               specimen in meters.
               Current value is     1.52400
          6   Distance between transmitter bar gage and the
               specimen in meters.
               Current value is     1.52400
          7   Bar wave velocity in meters/second
               Current value is    5183.70
          8   Density of the bar in Kg/meters cubed
               Current value is    7830.00
              Enter number of parameter to be changed, or 0 if
               you are done changing.
               8
              Enter new value
               7741.00
              Enter number of parameter to be changed, or 0 if
               you are done changing
               0
```

When all the changes have been made, or if none were needed, the initial part of the program is completed. At this point the user can choose between proceeding with the dispersion phase correction on the bar pulse data, or going directly to the data analysis for a specimen. User may wish to perform analysis on previously corrected data, or on present uncorrected data.

For the integer-number choice questions, if the user responds by typing an integer not on the menu, the question will be asked again. If the response is not an integer, the execution of the program will be aborted and must be started from the beginning.

Phase Correction:

User needs to input: (a) how many specimens he wants to analyze, (b) the ID name for each specimen.

```
EXAMPLE:  DO YOU WANT TO SKIP THE CORRECTION PHASE OF THIS
          PROGRAM AND GO DIRECTLY TO THE ANALYSIS PHASE? Y/N
          N
          All the input data are ready in units of microstrain
          and stored in the files of the required filenames?
          Y or N
          Y
          ENTER THE NUMBER OF SPECIMENS TO BE ANALYZED
          1
          ENTER SPECIMEN ID FOR SPECIMEN NO.
          W24
```

The correction of the pulses takes about three minutes per specimen on the PC-XT currently used, that is three minutes for a set of incident, reflected, and transmitted waveforms. When this task is completed the user can stop the program execution, or proceed to the data analysis phase.

```
EXAMPLE:  THE CORRECTIONS ON 1 SET(S) OF DATA HAVE BEEN
          CALCULATED.  DO YOU WISH TO PROCEED WITH THE
          STRESS AND STRAIN CALCULATIONS? Y/N
          Y
```

Data Analysis:

The user has to input (a) how many specimens need to be analyzed, (b) the ID name for each specimen, (c) specimen length in meters and (d) the area ratio of the bars to the specimen.

The program is then set up to calculate stresses, strain rates, strains, and velocities for each specimen. Since not all these quantities may be of interest every time, the user is asked before each calculation whether it is to be performed.

EXAMPLE:
```
Which set of data is to be used
Original (0)  or  Corrected (1)
1
Enter the total number of specimens
1
ENTER SPECIMEN PARAMETERS FOR EACH SPECIMEN IN THE
FOLLOWING:
Enter #   1 specimen ID
W24
Enter the specimen length in meters
0.0694
Enter the cross section area ratio of the bars to the
specimen
1
DO YOU WANT TO CALCULATE STRESSES?  Y/N
Y

W24ssf.dat
The above is the filename of the output file containing
specimen stresses at the incident and transmitter
interfaces, and their average. All are in units of MPa.

DO YOU WANT TO CALCULATE STRAINS AND VELOCITIES? Y/N
N
FINISHED AT LAST.  HAVE A NICE TIME.
```

SHPB is now completed. Outputs are stored in the following files:

IDif.dat, IDrf.dat, IDtf.dat contain the dispersion corrected waveforms for the incident, reflected, and transmitted pulses for a specimen.

IDsso.dat, IDssf.dat contain the calculated stresses, $(\sigma_1, \sigma_2, \bar{\sigma})$, for the original and the corrected waveforms respectively. (The f denotes Fourier for the correction by Fourier analysis.)

IDsno.dat, IDsnf.dat contain the calculated strain rates, strains and velocities, $(\dot{\epsilon}, \epsilon, V_1, V_2)$, for the original and corrected waveforms respectively.

2. INTRODUCTION AND GENERAL DESCRIPTION OF THE PROGRAM

Program SHPB performs wave dispersion correction and data analysis for a split Hopkinson pressure bar system. The program is divided in two main parts each dealing with one of the above tasks. Before starting either or both parts, an initialization process is needed. The parameters associated with the system in use are either read from a stored file or entered (or corrected) by the user interactively. The wave dispersion correction is performed using a Fast Fourier Transform (FFT) algorithm. The data analysis uses well known relationships for elastic media, to calculate the stresses, strains, velocities, and so on. This analysis can be done on the corrected or the uncorrected pulses.

In general this program is intended to be user friendly, all the inputs and decisions on what to calculate are done interactively, with simple menu choices. SHPB will automatically save the parameters for a system, as well as the phase angle corrections for future use, for as long as none of the parameters are changed. This saves time and reduces errors in the initialization process. A flow chart of SHPB is included at the end of this section to give a complete overview of the program.

The program is so arranged that it will display the name and parameters for the last system for which the program was used. The user is then asked whether the same system is to be used again. If the answer is yes,(Y), the user is further asked if any of the values need to be modified. If the same system is not used, a directory of the available systems appears on the screen. When one of these systems is chosen, the parameters pertinent to that system are shown on the screen. Whatever system has been picked, the user can further decide whether the values need to be modified or not. If they are to be changed, the user is prompted on the screen to input the corrected values. The bar diameter, the distances of the strain gages, and the bar wave velocity, are all input in meters or meters/second. These values are then stored in a file and will thus,replace the old values.

Furthermore, when new parameters are introduced, new phase angle corrections need to be calculated. This is done by calling subroutine 'ANGLE', and those values are also stored in a file for further use. The values stored are the phase shifts per unit distance between pressure bar gage and specimen interface. At this point the analysis can begin by reading in the input data and performing the phase corrections.

Each pulse is divided by equally spaced points. The interval between these points is the time step (tstep). The resulting number of points is called 'ni', and is one of the required input parameters of the program. In order to make the FFT algorithm efficient, the number of points to be analyzed for each pulse should be a power of two, e.g. $2^{10}=1024$. The SHPB program is written to accept any number of input data points (ni), up to 1024. If the number input (ni) is not a power of two, the program will select a power of two (nm) which is just large enough to include all the input points and will assign the value zero as data to points numbered above ni (up to nm). The program then chooses the base period (T) for the FTT analysis as T=(nm)*(tstep) where (tstep) is the time step in microseconds. The added zeros only appear in the FFT calculation. The output pulses are truncated at (ni) points.

The parameter 'mm' represents the number of the discrete frequencies that the Fourier transform will correct. Usually a number around 50 is more than sufficient, since higher frequencies do not affect the solution. The lowest frequency used by the FFT program is 1/T; the lowest circular frequency is $wo=2\pi/T$. Phase shifts are calculated for circular frequencies that are integral multiples of wo up to mm times wo.
Before the FFT subroutine is called, the incident, reflected and transmitted pulse data are read in (see program segment beginning with step 4), and the parameter dz is set equal to dzi,-dzi or -dzt according as k=1,2 or 3.

The FFT subroutine then does a frequency analysis of each pulse, determining the amplitude and phase of each of the mm frequency components. Each phase is corrected by adding to it a

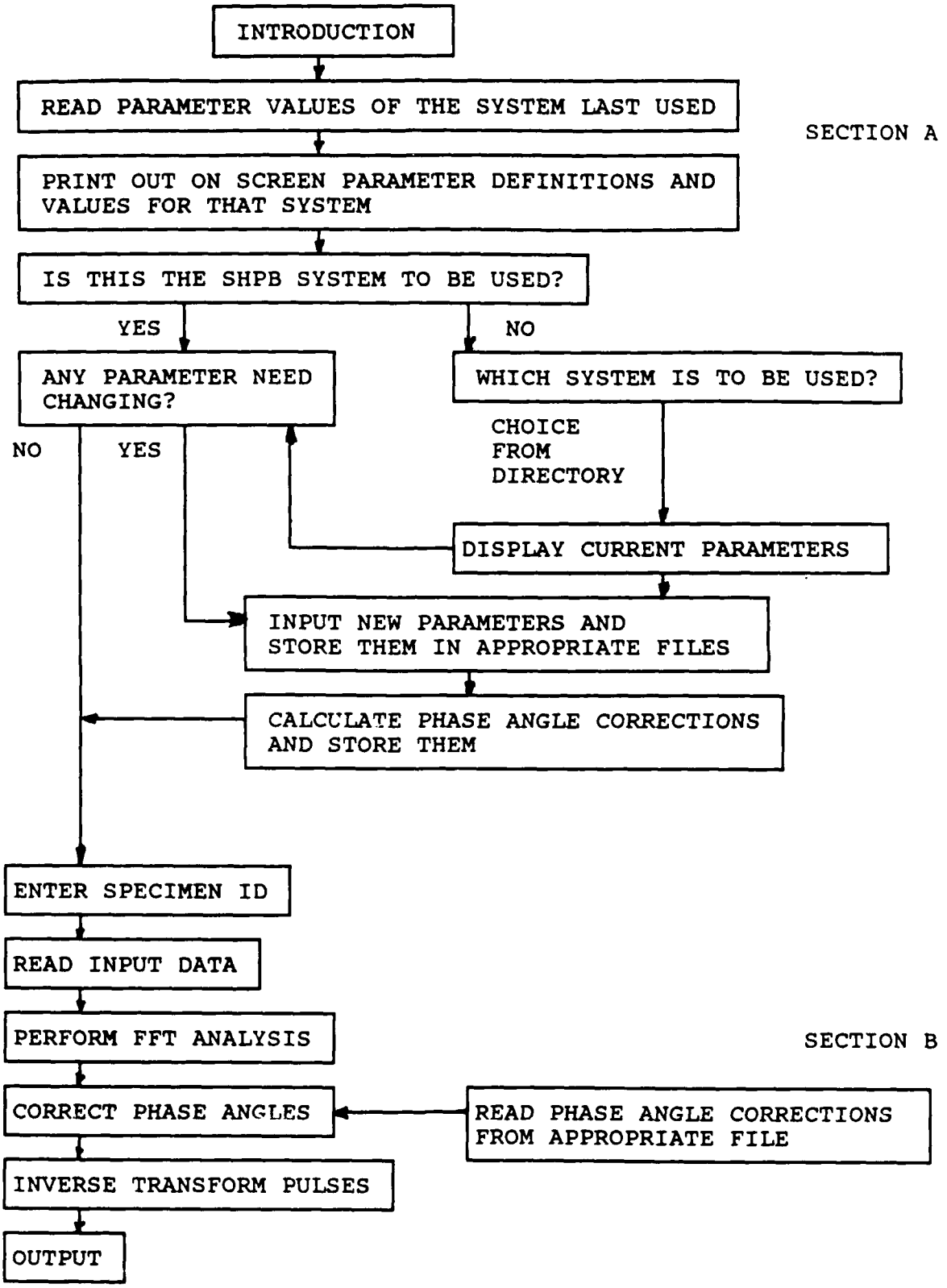
correction calculated by multiplying dz times the phase shift per unit distance that was calculated by subroutine ANGLE for that frequency component. Then the FFT subroutine reassembles the pulse as it appeared at the specimen interface.

At this point the first part of the program is completed. If the user wants to proceed to the specimen analysis part, he is asked for the specimen(s) ID, since he may not want to use the same specimens the dispersion correction was just done on. Also the area ratio of the bars to each specimen, and the specimen lengths, need to be entered at this point. The user is further asked if the corrected or the uncorrected pulses need to be analyzed. The data analysis itself is fairly straightforward and the user can interactively decide what quantities to calculate.

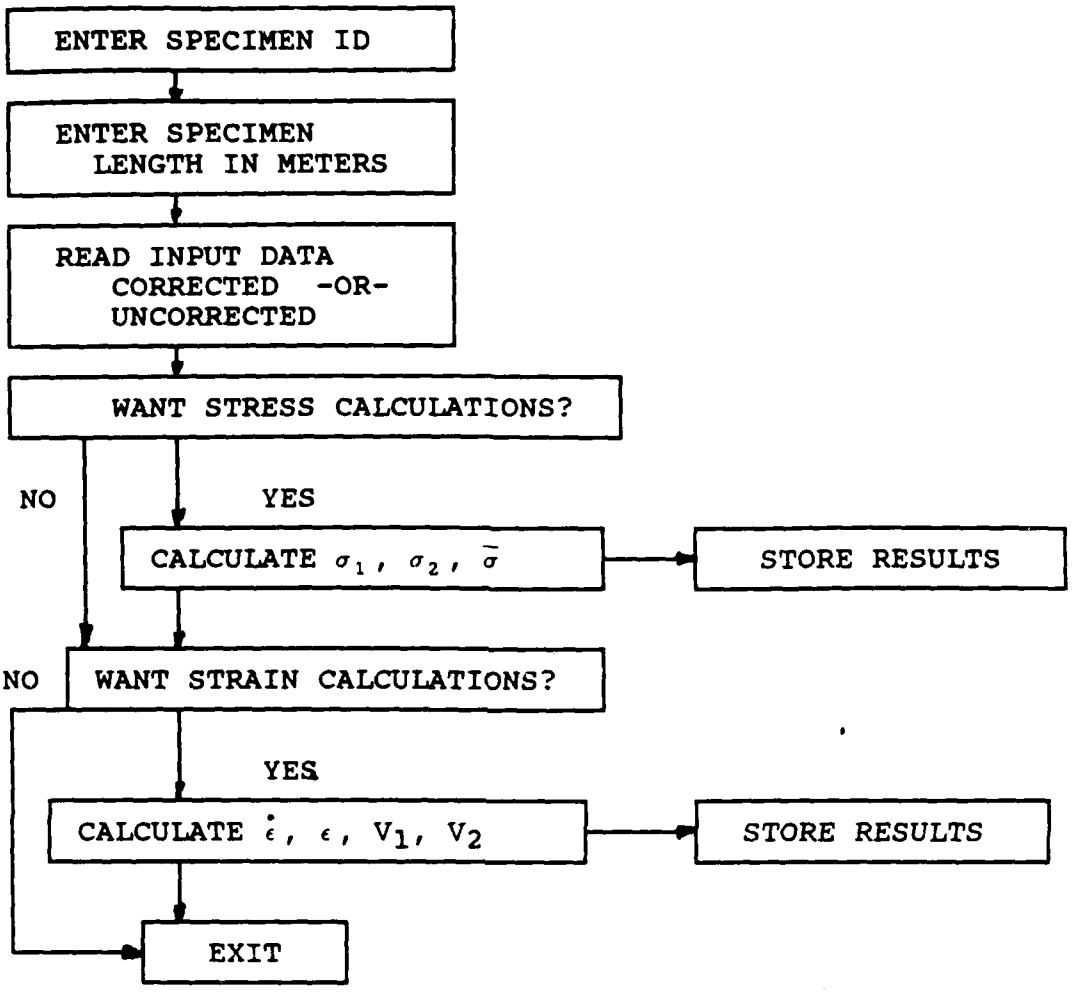
A description of all the output files generated by SHPB is included in Section 4. Program SHPB assumes that the input data are in formatted files in the computer in units of microstrain and leaves the output data in formatted files with the following units: stress in MPa, strain in percent, strain rate in S^{-1}, and velocity in m/S. Section 5 gives a brief account of how the input files are transferred into the computer from the Nicolet, and Section 6 describes the graphical output procedures currently in use at the University of Florida. Different input and output procedures will be used at AFESC/RDC.

The data analysis performed by SHPB can, in the future, be expanded to calculate other quantities that may be needed, such as energy. A very important addition needed is the incorporation of a plotting routine coupled with a suitable graphic package that would permit the display of the results. The procedures currently in use at the University of Florida, described in Section 6, may not be the ones selected for another system.

FLOW CHART


INTRODUCTION
READ PARAMETER VALUES OF THE SYSTEM LAST USED
SECTION A
PRINT OUT ON SCREEN PARAMETER DEFINITIONS AND VALUES FOR THAT SYSTEM
IS THIS THE SHPB SYSTEM TO BE USED?
YES
NO
ANY PARAMETER NEED CHANGING?
WHICH SYSTEM IS TO BE USED?
NO
YES
CHOICE FROM DIRECTORY
DISPLAY CURRENT PARAMETERS
INPUT NEW PARAMETERS AND STORE THEM IN APPROPRIATE FILES
CALCULATE PHASE ANGLE CORRECTIONS AND STORE THEM
ENTER SPECIMEN ID
READ INPUT DATA
PERFORM FFT ANALYSIS
SECTION B
CORRECT PHASE ANGLES
READ PHASE ANGLE CORRECTIONS FROM APPROPRIATE FILE
INVERSE TRANSFORM PULSES
OUTPUT

SECTION C


ENTER SPECIMEN ID
ENTER SPECIMEN LENGTH IN METERS
READ INPUT DATA CORRECTED -OR- UNCORRECTED
WANT STRESS CALCULATIONS?
NO
YES
CALCULATE σ_1, σ_2, $\bar{\sigma}$
STORE RESULTS
NO
WANT STRAIN CALCULATIONS?
YES
CALCULATE $\dot{\epsilon}$, ϵ, V_1, V_2
STORE RESULTS
EXIT

3. HOW TO USE THE PROGRAM

Before program SHPB can be run, the following files MUST be present in the same directory, properly formatted and in units of microstrain:

(1) IDi.dat, IDr.dat, and IDt.dat

These are input files, they contain incident, reflected, and transmitted data for a specimen with an ID name, which consists of one to three characters (numbers and/or letters). In order to create these files properly, the user needs to treat the raw data from the SHPB experiment in the following way: the data needs to be normalized to the zero point, converted to microstrain through calibration factors and appropriately formatted.

The data for all three pulses needs to be in format f10.3, with one point per line, and a separate file for each pulse is required. Section 5.2 gives possible procedures involved in the data transfer from the Nicolet oscilloscope to the computer, which also normalize and convert the transferred data into microstrain.

(2) SYSTEM.dat

These are one or more files containing information pertinent to a particular system. Currently the following four files are available: UF.dat, TYND.dat, UF2.dat, and NEWSYS.DAT. The parameters for the University of Florida 3-inch SHPB system, Tyndall 2-inch and University of Florida 3/4 inch SHPB system are stored in the first three, respectively. NEWSYS.DAT is a file where all the parameters are currently zero. If number 3 is chosen from the system directory, they will appear as zeros. The user can then change to the proper values, as indicated in the User's Guide. There is another file, called LASTSYS.DAT, which records the number of the last system used (1, 2, 3 or 4). When SHPB is started this is automatically called and used by the SHPB program to display the parameters of the last system used. These five files are on the program diskette and should be stored on

the hard disk of the user's computer. When the program is run, it will retrieve one of these files, (the one for the system that was used the last time), read the content, and display it on the screen. If the user is using this program for the first time, and is not sure how to input the above files, he can just create them and leave them empty. The program will read all zero's, then ask the user to input new parameters. SHPB will then automatically write to these files, with appropriate formatting, for future use. The stored parameters are the following, in order (format shown in parenthesis): time step (f7.5), diameter of the bar (f7.5), distance between strain gage on the incident bar and the specimen (f7.5), distance between the strain gage on the transmitter bar and the specimen (f7.5), wave velocity in the bar (f7.2), number of discrete frequencies that will be corrected (i4), number of points to be analyzed (i4), and density of the bar (f8.2).

(3) Angle.dat, Tangle.dat, Tuf2.dat

These files contain the dispersion phase angle corrections calculated for the UF 3-inch system, the Tyndall 2-inch system and the UF 3/4-inch system, respectively. Every time any of the parameters are changed, the corrections are recalculated, and the new values will replace the old ones in the above files. The first time one is running SHPB, if the angle files are not already available, the user can modify or reinput any of the parameters, to "force" the program to create and write to the needed file. Once these files are present, the program can be run.

The program title and the parameters of the system last used are shown on the screen. The user is then asked whether that is indeed the system he wants to use and/or if the parameters need changing. Once a system has been chosen, the parameters have been set, and new phase angle corrections per unit distance have been calculated, program SHPB is ready to perform the dispersion phase correction for the bars followed by specimen data analysis,

or just the dispersion correction, or just the specimen data analysis. The user is prompted with Y/N questions for these choices. Within the specimen data analysis part, the user has further options on what is being calculated, that is stresses, stresses and strains, strains only, etc.

If data for more than one specimen are being analyzed, SHPB will perform the phase corrections on ALL of them before entering the data analysis part of the program. The user is asked again, at that point, how many and which specimens will be used, so that stress and strain calculations aren't necessarily done on all the corrected data, if any at all.

4. OUTPUT FILES

All files, input and output, start with the ID of the specimen. The dispersion correction part of the program produces three output files containing the three corrected pulses; (incident, reflected, and transmitted). They are:

IDif.dat, IDrf.dat, and IDtf.dat

Format is the same as inputs, that is f10.3. The specimen data analysis part of the program reads either the original input waveforms or the corrected ones from IDi.dat or from IDif.dat respectively (and similarly for the reflected and transmitted pulses). The output for this section contains stresses, strains, and velocities for each specimen analyzed. They are stored in the following way:

- Files IDsso.dat and IDssf.dat contain the stresses for the original and the corrected data respectively.

- Files IDsno.dat and IDsnf.dat contain the strains and velocities for the original and corrected data respectively.

5. DATA TRANSFER FROM NICOLET TO PC

5.1 Introduction

Program SHPB analyzes data files which are assumed already present in the computer memory or on diskette files that can be called by the program. Before the program can be used it is necessary to transfer the data into the computer from the Nicolet 4094 digital oscilloscope that was used for data acquisition. The Nicolet system is equipped with diskette drives which can store the data along with some normalization factors on floppy disks, but these floppy disks are not compatible with the usual diskette drives of the IBM-XT type of PC. Data transfer is, therefore,currently being done at the University of Florida by cable through the RS-232 interfaces on the Nicolet and the PC.

The IBM-AT PC at AFESC/RDC is equipped with a drive that is compatible with the Nicolet floppy disks. It can,therefore, transfer the files directly into the PC from the Nicolet diskettes. Some use will need to be made of the normalization factors (which are stored in the first records of each diskette file). It will also be necessary to name and format the treated transferred files, so that they are in the proper form to be called and used by program SHPB.

5.2 Data Transfer Through RS-232 Interfaces via PDP-11

The transfer procedure described here is the first one developed in the Engineering Sciences Department laboratory at UF, called S-1 in the March 1987 progress report. It uses a FORTRAN program in the DEC Micro PDP-11 to control transfer of the data files from the Nicolet memory into the memory of the PDP-11, using the RS-232 interfaces. Then a communication package (PC-VT) is used to transfer the files from the PDP-11 to the PC, again using the RS-232 interfaces. No use is made of the normalization factors stored in the first few records of each Nicolet diskette file, since, when Program S-1 was written, it was not clear how to interpret the stored normalization factors.

Instead of using the stored values, the needed factors are manually determined from visual observations of the Nicolet screen and dial settings. The default voltage range setting was chosen as two volts for the full Nicolet screen. The program prompts the user to input manually any change in the voltage range, and then computes the correct factor.

At the beginning of the data transfer the initial start time of the incident pulse is determined from the Nicolet scope display. If strain gages mounted on the specimen are to be recorded in addition to the SHPB pulses, the beginning of the SHPB transmitted pulse is determined from the scope display for comparison purposes; this start time of the transmitted pulse is not used in the SHPB analysis. Initial start points for the pulses from gages mounted on the specimen has been chosen as the arithmetic mean of the incident and transmitted pulse start times.

Eighty points preceding the determined initial start time, also recorded from the Nicolet screen, are used to establish a base line zero for the vertical display, by averaging the voltages of these 80 points. Each individual point of both the incident pulse and the reflected pulse is then referred to this base line. Another zero base line is similarly determined for the transmitted pulse, since it is recorded on a different channel.

The initial start time for the reflected pulse at the specimen interface is determined by adding dzi/C_o to the observed incident pulse beginning. Here dzi is the distance from the incident-bar gage station, and C_o is the bar-wave speed. For purposes of SHPB analysis this same start time is also used as the beginning of the transmitted pulse at the second interface, although the signal does not actually begin there until a slightly later time after transmission through the specimen.

A pulse length of 511.5 microseconds (1024 points at 0.5 microsecond intervals) has been chosen for each pulse in the UF 3-inch SHPB system, including strain pulses from the specimen-

mounted gages. This is a little longer than is needed to record all the useful data points of each pulse.

The determined initial start times of the incident and the transmitter pulses have been offset by the initial beginning time of the screen in order to have correctly corresponding numbers for the start points of these two pulses.

During the testing of data transfer several data points were randomly chosen to find out the factor between the voltage readings on the Nicolet display screen and the associated transferred data. The factor so determined is 16 for the two-volt voltage range setting. This number was then implemented into the transfer program and the voltage range setting of two volts has been used as the default setting. When the transfer is initiated, the program prompts the user for factors for each channel, which are the ratio of the used voltage range setting and two volts.

The strain gage factors used in the incident and transmitter bars were incorporated into the program to convert the transferred data from voltages to microstrain. Gage factors used for gaged specimens have been input by the user in response to prompts.

For each tested specimen three pulses, i.e. incident, reflected and transmitted pulses, in units of microstrain, with the same pulse length, have been acquired in running the transfer program. Additional pulses have been transferred for gaged specimens. All pulses have been formatted in FORTRAN real variable format f10.3.

5.3 Future Plans for Data Transfer

As described in the June progress report, a BASIC language program has been written to transfer files directly from the Nicolet to the PC through the RS-232 interfaces without using the PDP-11 as intermediary. The transferred files contain in the first few records of each file certain normalization factors, such as Vnorm, Hnorm, Vzero, etc. These factors are defined and

described in the Input/Output Section of the Operations Manual for the Nicolet Series 4094 Digital Oscilloscope. It should be possible to use these normalization factors to get the input pulses in terms of voltages without the manual input of the voltage range settings described in Section 5.2, but we have not yet verified that our procedure works. Verification is delayed by an equipment failure in the Nicolet. When that has been repaired, and the procedure verified, a substitute Section 5.2 will be provided for the Handbook.

The procedure of Section 5.2 will not be used at AFESC, where direct input by inserting the Nicolet diskette into the IBM-AT computer will be possible. When that procedure has been verified at AFESC, it may be desirable to add a short description of it to the Handbook.

6. GRAPHICS OUTPUT

6.1 Introduction

The SHPB program leaves the results in output files in the computer, which can be printed out and/or stored on diskette. It is important to have available some graphics software and hardware to produce suitably labeled high-quality plots of the results. These may vary from one system to another. For high quality plots, a plotter or laser printer is needed. It is desirable, therefore, to have graphics software that is capable of controlling output to these devices. Almost any graphics software can control output to a dot-matrix printer, but not all are convenient for driving a plotter.

Some desirable features for the graphics software are:

1. It should be able to produce plots with high resolution, using a large number of points.
2. It should have suitable fonts for symbols to label the coordinate axes, legends, and captions.
3. It should be possible to re-scale the figure to fit in a suitable space on the page, after it has once been plotted on the screen.

4. It should be possible to scale, position and edit different parts of the legend and captions by on-screen editing.
5. It should be able to control a plotter and also a laser printer.

At the University of Florida these requirements have been met by using two compatible graphics software packages. There may be one package available that will do it all.

Lotus 1-2-3 Symphony is an upgraded version of Lotus 1-2-3, a commercial work sheet software package. In addition to various bar charts and pie charts, it can produce a high-resolution line graph, by plotting up to 4000 points. (The original Lotus 1-2-3 also works.) It is not, however, designed for scientific and engineering work, and it does not have all the desired on-screen editing capability.

Freelance Plus is another graphics package, produced by Lotus, which is fully compatible with Lotus 1-2-3. Freelance Plus by itself satisfies all the requirements listed above except the first one. Its graphs are limited to no more than 120 points each. It is not really designed for engineering and scientific work either, but it has all the fonts needed for symbols and it has the capability of scaling the drawing, and of separately scaling, positioning and on-screen editing the text that appears with the drawing. It can draw arrows to special points on the graph and label them. It can drive our HP-7470A plotter.

It is possible to draw the curve with Lotus 1-2-3 and save it in a disk. Then the file is recalled under control of Freelance Plus. The recalled high-resolution plot can then be scaled and labeled and output to the plotter. It may be possible to produce the high-resolution plot by some other graphics package and still recall the file under control of Freelance Plus, but we have only verified it with Lotus 1-2-3. An example of graphics output done this way is given in Figure 1 at the end of the appendix.

6.2 Graphics Procedures with Lotus 1-2-3 Symphony and Freelance Plus

The following is a procedure of operations on Lotus 1-2-3 Symphony and Freelance Plus for a plot of output data from the program SHPB. The user, even if he is a beginner with the two software packages, can produce a satisfactory plot as long as he follows the procedure step by step. The plot consists of no more than six smooth curves with a major title, a subtitle, and labeled horizontal and vertical axes. The plot may be located anywhere and of any size the user chooses on the page. The procedure gives relevant explanations as well as the details about what keys to press and what to enter (i.e., type and then press RETURN). Once the user is more familiar with the features of the software, he will be able to change some steps for his particular demands.

I. In Lotus 1-2-3

1. Enter 123 at the prompt in the 1-2-3 subdirectory. (This means type 123 and Press RETURN).
   Soon the screen displays the blank 1-2-3 worksheet in READY mode.
2. Press the slash key (/).
   The worksheet comes to MENU mode. The screen displays a command menu (a series of command choices) on the control panel (the top part of the screen).
   To "Select" menu items, move the menu pointer (cursor) to the choice and press RETURN, or type the first letter of the menu item.
3. Select FILE-IMPORT-NUMBER (Select FILE in the menu and then IMPORT and then NUMBER).
   Select from the displayed file list or enter the filename whose data are to be plotted. The screen displays the beginning rows of the data filling in the worksheet. Each variable (for example, the stress at the specimen incident interface) occupies one column.

Each row stands for an instant of time, which precedes the underneath row by the time step (tstep in Program SHPB).

4. To import another set of data to the worksheet, first move the cursor to the upper left corner of the blank part of the worksheet, then repeat Step 3.
5. If some of the data is to be plotted versus time, repeat Step 4 to import the time for each row from a previously prepared file TIME.DAT.
6. To save the worksheet in the subdirectory 123, Press /, and then Select FILE-SAVE. Enter a filename when prompted.
7. To choose the horizontal axis range, Press /, and Select GRAPH-TYPE-LINE-X. Enter the horizontal axis range. For example, if the data in column E are to be plotted on the horizontal axis, with the range from the datum in cell E1 to the datum in cell E1024, enter E1..E1024.
8. To choose the vertical axis range for one curve, Select A or B or whatever column contains the data to be plotted. Enter the vertical axis range.
9. To provide more than one curve on the same plot, repeat Step 8. Lotus 1-2-3 is able to plot up to 6 curves.
10. Select OPTION-FORMAT-GRAPH-LINES.
11. Press ESC twice. Select OPTION-SCALE-SKIP. Enter the skip factor. For a skip factor n, Lotus 1-2-3 plots every nth entry from the x range. The skip factor n should be large enough to make the labels on the horizontal axis spaced at appropriate intervals.
12. Select SCALE-Y SCALE-LOWER. Enter the Lower Limit of the plot in the Y direction.
    Select UPPER. Enter the upper limit of the plot in Y direction.
    (This step is to arrange the plot in a frame of an appropriate height. It is not a mandatory step.)

13. Press ESC twice. Select TITLE-FIRST and enter the plot title. Select TITLE-SECOND and enter the subtitle. Select TITLE-X AXIS and enter description (variable names and units) for the horizontal axis. Select TITLE-Y AXIS and enter description for the vertical axis. (Omit this step, if the user does not want titles for the plot.)
14. Select QUIT-VIEW.
    The screen displays a plot. Rerun the above steps if the plot is not satisfactory.
15. Press /. Select SAVE (It is GRAPH SAVE, not FILE SAVE). Enter a filename to store the graph's image. A default extension .PIC would automatically be given. That saved graph cannot be brought back to the screen under control of Lotus 1-2-3, but can be recalled when Freelance Plus used.
    When the file (filename.PIC) has been generated and saved, the phase in Lotus 1-2-3 is completed.

NOTE: To change the worksheet from the READY mode to MENU mode, press /. To recall the last menu, press the escape key (ESC).

II. In Freelance Plus

1. Type FL at the prompt in the FLP subdirectory. Press RETURN, RETURN, RETURN. The command menu appears.
2. Select FILE-IMPORT-PIC. Enter the filename.PIC. Wait a while. The screen displays the same plot created and saved in Lotus 1-2-3.
3. The commands in the menu give complete freedom to edit, rearrange the current plot and so on.
   Some examples are as follow:
   (1) Changing the plot size: Select SELECT-ALL-EDIT-SIZE-UNIFORM. Move the cursor. The size of the whole plot changes (reduces or enlarges).

(2) Turning the plot: Select REARRANGE-TURN. Move the cursor. The plot turns by an angle the cursor turns by.

(3) Moving the plot: Select MOVE. Move the cursor. The plot moves.

The double page feature and symbols would further help to perfect the plot.

4. Select FILE-SAVE. A default extension .DRN would be given.
5. Select PLOT-GO. The HP-7470A plotter plots out the graph. (The plotter is connected to the computer through the RS232 interface.) An example is shown in Figure 1 at the end of the appendix.

7. ENGINEERING AND SERVICES LABORATORY SYSTEM

7.1 Introduction

The previous sections of this handbook were given as a general description of the dispersion corrections and transfer procedures for the University of Florida system. This section is written specifically for the users of the Engineering and Services Laboratory (ESL) system. The ESL system consists of a 2.0 in (51 mm) SHPB, a 4094 Nicolet Oscilloscope with floppy disk storage, and an IBM AT 33A PC with VUPOINT and LOTUS software packages. The following sections describe in detail the use of the ESL system.

7.2 General Description

In order to make the SHPB data analysis as expedient and user-friendly as possible, two subroutines have been added to the existing software. The first subroutine (prep), "prepares" the data for use by the main procedure of SHPB. The waveforms as recorded by the Nicolet oscilloscope are in volts and VUPOINT stores them with the associated times and other information, in specified files. The function of PREP is to read these files, discard the first eleven lines of title, dates, and other general information, convert to the correct units and store the amplitude values in a format compatible with SHPB (F10.3). Further, since the recorded pulses vary in length, the data needs to be adjusted so that the number of points is a power of two (2); thus PREP adds as many zeroes as needed to reach, for example, 1024 points.

SHPB was written to handle compressive data, meaning that the incident and transmitted pulses are assumed positive and the reflected negative. Therefore, when dealing with tensile test data, depending on how the waveforms were recorded, some or all need to be inverted. PREP will do this at the user's command. The second subroutine (SEC) reads the appropriate data from the SHPB output and sets up files that VUPOINT can interpret and plot. Both subroutines are menu-driven, and interactive so that

the user is "guided" through the entire procedure. After each part the user can exit the program, or skip a part. In all this, real time loses its meaning, the program only keeps track of the number of points and assigns an initial representative time of 0.0 seconds and it increments this initial time by the given time step. Three command files were written to aide in the time consuming and tedious task of transferring files from one directory to the other and calling the correct programs at the right time. The first command file is called FAST and it resides in the VUPOINT directory. It automatically transfers the three pulses, incident, transmitted, and reflected, from VUPOINT, (where they have been recorded from the 4094 Nicolet disk), to the SHPB directory. It calls the program TOTAL (SHPB with SEC and PREP included), and it transfers the output plot files back to VUPOINT. In other words, the command file FAST takes the user from VUPOINT, through all the data manipulation needed, and back to VUPOINT to view the graphical output.

The second command file is called TRANSFER, and it will send three files at once from SHPB directory to VUPOINT. The names of the files, WITHOUT extension, must follow the command TRANSFER. This file is found in the SHPB directory.
The third command file is called PLOT and it transfers ALL plot files for one specimen from the SHPB directory where they are created, to the appropriate directory where they can be displayed. VUPOINT and LOTUS are automatically called.
Since VUPOINT will only plot amplitude values versus time (or versus any equally spaced data), stress versus strain curves cannot be displayed. As these are an important element of any material testing analysis, another graphic package had to be considered. The first and most obvious choice was the program LOTUS which is the one used (in part), by the University of Florida in Gainesville. Although LOTUS is not designed for scientific or engineering work, it will produce high quality plots with up to 4000 points. It is able to read the data as calculated by SHPB and it can plot stress versus strain data. It also gives flexibility in formatting the plots, rescaling, labeling

axes, legends, and captions, and other useful features all in an interactive user-friendly environment. Furthermore it will drive the HP 7475a plotter and the OKIDATA printer. Lotus is now part of the main menu and the command file PLOT will transfer the stress/strain data directly to LOTUS instead of VUPOINT. A separate, small command file called TR will transfer just the stress/strain data to LOTUS, and run the program.

It is important for the user to understand that, to prevent the accumulation of obsolete files, and yet maintain the flexibility to store, and analyze data, some files, specifically all the files in VUPOINT, are OVERWRITTEN each time a new set of data is retrieved. It is imperative then, to process this data through program TOTAL (new version of SHPB), before obtaining new data from the 4094 disk. Files with the specimen ID associated with their names will be stored until the user deletes them. These include the original and corrected pulses, and the stresses and strains information. Since the files containing the data to make the plots are not unique to a specimen, what does not get converted to a VUPOINT plot and printed out, will be lost. These files make very inefficient use of disk space and it would therefore be unpractical to save nine of this kind for each specimen; if, later in time the user wishes to retrieve the stress curve for specimen x, the program SEC can be run, and the needed plot reproduced; this should only be a short procedure and this way the available memory will not become quickly saturated with superfluous data. As a last note, the files stored under the specimens' ID names in subdirectory SHPB, may with time reach a confusing number. It may be advantageous to copy them to disk and clear the permanent storage at regular intervals.

7.3 QUICK REFERENCE GUIDE

BASIC PROCEDURE:

- Use VUPOINT to get one set of data (three waveforms) and store them in INC.ad, REFL.ad, and TRANS.ad.
- Exit VUPOINT and type FAST. Follow instructions. Command file will bring the user to LOTUS and then back to VUPOINT

after the three waveforms have been corrected, manipulated accordingly, and the stresses and strains and the plots have been calculated.

- Use LOTUS to see, save, and plot stress/strain curve.
- Use VUPOINT to look at curves. Make a permanent plot for the ones to be saved/printed out.

WITHIN VUPOINT:

- To read/look at data select INPUT
- To save data as a permanent file select OUTPUT and WRITE-DOS-FILE
- To plot data need to create a PLOT file which can then be plotted. To do this select OUTPUT and PLOT. Answer questions, then select OUTPUT and DRAW. (Only trick there is to choose parity=ODD).

WITHIN LOTUS:

- To import/look/create a graph select 1-2-3
- To get a hard copy of a graph select PRINTGRAPH
- To get a menu display when in "ready" mode, press /
- To recall the last menu, press escape (ESC)

7.4 ESL SYSTEM USERS GUIDE

This portion of the User's Guide describes how to get data from the Nicolet oscilloscope's 4094 disk to the IBM PC, how to treat the data once it's stored in the computer, and how to obtain graphical results.

The first thing the user needs to do is to insert the disk from the oscilloscope with the data into drive A, run the program VUPOINT, select INPUT, then 4094-DISK, to read the data into the computer. Select the needed record and let VUPOINT display the waveform for set #1 on the screen. (Set #1 is the one representing the incident and the reflected pulses). Notice that although the waveforms are displayed it does not mean they are stored in the computer; the data is still only in memory and it will be lost if you exit from VUPOINT. Before storing the pulses

the user must find the initial and final times of each one: press TAB and "return", twice, to enlarge the display. Move the cursor to your choice of the beginning of the incident pulse (approximately -50 us), and calculate the start of the reflected pulse (and thus the end of the incident wave), by adding a time equal to twice the distance of the specimen to the strain gage divided by the bar wave speed, and record these values. For the transmitted pulse use the same initial and final times as the reflected. Now the user is ready to save the data; press ESC,(*) select OUTPUT, then WRITE-DOS-FILE. Press "return" twice and choose file INC.ad to write the incident pulse data. Input the correct times, press "return" for the other questions(use default values). Select OUTPUT again, and repeat the above process, from (*), twice; once by writing the reflected data to REFL.ad and once by writing the transmitted data to TRANS.ad. The data for one specimen is now stored in the computer. Select EXIT.
Once out of VUPOINT the user should see:

```
C:\VUPOINT>
```

Type in FAST. The following will appear on the screen:

```
C:\VUPOINT>fast
C:\VUPOINT>copy inc.ad \shpb\inc.ad
1 File(s) copied
C:\VUPOINT>copy refl.ad \shpb\refl.ad
1 File(s) copied
C:\VUPOINT>copy trans.ad \shpb\trans.ad
1 File(s) copied
C:\VUPOINT>cd\shpb
C:\SHPB>total
enter specimen id
```

Here the user needs to input the id name/number (three characters) that will identify this set of data from now on. This is the beginning of the revised SHPB program (called TOTAL) that treats the data appropriately and sets up the right files for the main (old) part of SHPB. The user needs to treat ALL THREE pulses. They all need a magnification factor, some, as explained in the manual, need to be inverted.

Example: ("u' preceding a line indicates user's input)

enter specimen id

u... p05

is this an incident pulse? Y/N

u... y

enter calibration factor, microstrains/volts

DON'T FORGET TO INCLUDE MAG. FACTOR!!!

u... 70.0

is this a tensile test? if the incident

pulse is NOT positive you

need to invert the waveform(s). Invert? Y/N

u... y

Have you prepared all three pulses? Y/N

u... n

is this an incident pulse? Y/N

u... n

is this a reflected pulse? Y/N

u... y

enter calibration factor, microstrains/volts

DON'T FORGET TO INCLUDE MAG. FACTOR!!!

u... 70.0

is this a tensile test? if the incident

pulse is NOT positive you

need to invert the waveform(s). Invert? Y/N

u... y

Have you prepared all three pulses? Y/N

u... n

is this an incident pulse? Y/N

u... n

is this a reflected pulse? Y/N

u... n

is this a transmitted pulse? Y/N

u... y

ETC... the above procedure can be repeated as many times as needed, if a mistake is made on the reflected pulse, for example, one only needs to go back to the "is this a reflected pulse?"

question. When all three waveforms have been treated correctly an answer of "y" to "have you prepared all three pulses?", will bring the user to the beginning of the main procedure of SHPB. This used to be the beginning of the program and it displays the title and parameters for the system last used. This part of the procedure has not changed and the user can refer to the original User's Guide for information. Program SHPB only performed the phase corrections and calculated the stresses and strains. A new subroutine has now been added that will automatically read the stresses and the strains and set up plot files to be used by the graphics routine. The output on the screen will look like this:

```
DO YOU WANT TO CALCULATE STRAINS AND VELOCITIES? Y/N
u... y
FINISHED WITH STRESS AND STRAIN CALCULATIONS
At this point you can either exit this
program and use program SEC to prepare the
plots you desire to see.  Or you can stay with
this program which will set up all possible
plots.  Time step will be 0.5e-6 sec.
EXIT? Y/N
u... n
```

By answering "n" the program will set up nine plot files. Then the files are automatically transferred to the appropriate directories, and VUPOINT is called. VUPOINT is used to view, store, and print out seven of the nine files. The eighth and ninth file are the ones containing the stress/strain data, (stress 1 and stress 2). To view, store, or plot these files the user needs to use the program LOTUS located in LOTUS subdirectory. LOTUS and VUPOINT are automatically called when program TOTAL is finished. By answering "y" to the EXIT? question the program TOTAL will be terminated. At this point the stress and strain data for the specimen being analyzed are stored on the hard disk but no plot file exists. To set up one or more of these files, the user needs to run program SEC located in subdirectory SHPB. To do this, from any directory type:

```
u... cd\SHPB
```

The user should see:

C:\SHPB>

Type SEC, the following should appear on the screen:

WHAT DO YOU WANT TO PLOT?

1 corrected pulses vs time
2 stress 1 vs time
3 stress 2 vs time
4 strain vs time
5 stress vs strain
6 avg. stress vs time
7 strain rate vs time
8 velocity 1 vs time
9 velocity 2 vs time

ENTER CHOICE NUMBER

Every time a choice is made the output file name will be displayed on the screen. User needs to recall the file names in order to transfer them, later, to VUPOINT or LOTUS. Program will go back to main menu as many times as needed by answering "n" to the question "Do you want to quit this program", displayed after each plot file is created.

When program SEC is terminated, all files except the stress vs strain data, need to be transferred to VUPOINT. This can be done quickly by using the command:

TRANSFER FILE1,FILE2,FILE3

Three files can be transferred at one time, the file names should be separated by commas. To transfer the stress strain data to lotus, use the command TR. This will also automatically call LOTUS. If at any other time one wishes to use LOTUS or VUPOINT, from any directory type:

u... cd\

u... C:\MENU

and select appropriate one from the menu.

7.5 Graphics Procedures with LOTUS

LOTUS is an interactive, multipurpose program used here only to produce stress/strain curves. To use it follow these steps.

From main menu select LOTUS. Once in LOTUS select 1-2-3 (default), press return, then the slash key (/); a menu will appear on top of the screen. To select a menu item move the cursor to the choice and press return, or type in the first letter of the menu item. Select FILE, then IMPORT, then NUMBER. In the blue highlighted region the file names STST1.PRN, and STST2.PRN should appear; they are the ones containing the stress and strain data. Choose one and press "return". The screen will now display the beginning rows of the data, the strain occupies column A, and the stress column B.

Press /, select GRAPH, then TYPE, then XY, then X: for x-axis range enter a1..a1024, press "return, select A: for first data range enter b1..b1024, press "return". Select OPTIONS, then FORMAT, press "return" twice, then ESC. Select COLOR, press ESC, select VIEW. Graph will be displayed. Press ESC, select SAVE if plot is all right, OPTIONS if some changes are needed, (for example to put labels on axes, change the grid, change the scale, etc...). When the plot is complete, select SAVE, press "return", and either choose one of the highlighted names (replace an existing file), or type a new name. (The extension PIC will be automatically added).

User can now QUIT, press /, and QUIT (again), answer yes. Screen should be back to main menu. Select PRINTGRAPH to get a hard copy of the plot just prepared. Select IMAGE-SELECT (default), choose file to be plotted, (need to put # in front of the name with the SPACE bar), select GO. Make sure plotter is on, and paper is ready. Select EXIT when done.

NOTE: to get a menu display when in "ready" mode, press /. To recall the last menu, press the escape key (ESC).

7.6 Graphics Procedures with VUPOINT

VUPOINT is an interactive, data processing program designed to analyze, modify and plot data recorded by waveform digitizers; it will perform many operations (integration, curve fit, scaling, etc...), on time based data. To view the results of the SHPB program the following steps can be followed.
Select INPUT, then C, then the data file to be plotted. Choices are:

st1t.ad => stress 1 vs. time
st2t.ad => stress 2 vs. time
avst.ad => average stress vs. time
stnt.ad => strain vs. time
strat.ad => strain rate vs. time
vel1.ad => velocity 1 (in) vs. time
vel2.ad => velocity 2 (out) vs. time

Once the selected file is highlighted, press "return" three times, move cursor to A (actual data starts in line one), press "return" four times; the graph will be displayed on the screen. Press ESC for menu. At this point there are two basic choices: look at something else or save the data in a plot file that can later be printed out. To look at something else select INPUT, and repeat above procedure. To save the data in a plot file, select OUTPUT, then PLOT, then the "set #" to be saved, press "return" twice (default initial and final times), select C (default also); on top of the screen the following line appears: "select plot parameters to change for: New plot". This gives the user the opportunity to change axes labels, bounds, line style of a trace, titles, etc... Within the SHPB analysis usually only the y-axis label needs to be changed from volts to stress, or strain, with the right units. To do so, select AMP-AXIS, then LABEL, make the changes, select NO CHANGE to see the modified plot. Press ESC and answer yes if satisfied with the display; give the plot a meaningful name.
VUPOINT's memory is divided in four parts. Each set of data uses two, so that two sets of data can be kept in memory at one time.

This is useful when two curves need to be plotted on the same graph. To do this INPUT the two data files to SET # 1,2 and 3,4 respectively, select OUTPUT, then PLOT and save one set as described earlier. After this is done, and the plot file has a name, select OUTPUT, (again), and ADD-A-TRACE. This will allow the addition of a new data set to the previous plot. Axis-labels can be changed and captions can be added on the graph by typing K when the display is on the screen. To get a hard copy of the plot: from main menu select OUTPUT, then DRAW. Answer questions. Be sure plotter is on, choose ODD parity, 9600 baud rate, and 1 stop-bits.

```
c revised October 1987
c.....................................................................
c
c        MAIN PROGRAM FOR WAVE DISPERSION CORRECTION AND
c        DATA ANALYSIS IN A SHPB SYSTEM.
c.....................................................................
      DIMENSION   X(1024), Y(1024), XX(56), AG(56)
      DIMENSION  CSP(1024),XC(56),YC(56),AGC(56)
      DIMENSION  BAR(1200),PH(100),ar(100)
      dimension binc(1200),bref(1200),btra(1200),sl(20)
      character*8 idi,idr,idt,name
      character*10 idsts,idstn,ideng
      character*9 idif,idrf,idtf
      character*3 id,n(20)
      CHARACTER*9 RTITLE
      CHARACTER*5 OP
C
C     THE PROGRAM TITLE AND MAIN FUNCTION ARE PRINTED ON SCREEN
C
 150  format(i3)
 18   FORMAT(4f7.5,f7.2,2i4,f8.2)
      write(*,*)'THIS IS PROGRAM SHPB. IT PERFORMS THE PHASE '
      write(*,*)'CORRECTIONS FOR DISPERSION ON ANY WAVEFORM'
      write(*,*)'AND DATA ANALYSIS'
      write(*,*)'                                  '
C
C     CHECK TO SEE WHICH SYSTEM WAS USED LAST
C
      open(12,file='lastsys.dat',status='old')
      read(12,150)kk
      close(12)
 25   write(*,*)'THE PROGRAM PARAMETERS ARE CURRENTLY SET TO   '
      if(kk.eq.2)then
      open(8,file='tynd.dat',status='old')
      read(8,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (8)
      WRITE(*,*)'THE VALUES ENTERED FOR TYNDALL 2-inch SHPB'
      endif
C
      if(kk.eq.1)then
      open(10,file='uf2.dat',status='old')
      read(10,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (10)
      write(*,*)'THE VALUES ENTERED FOR THE U OF F 3/4 in. SHPB'
      endif
      if(kk.eq.3)then
      open(11,file='newsys.dat',status='old')
      read(11,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close(11)
      write(*,*)'THE VALUES ENTERED FOR NEW SYSTEM OF YOUR CHOICE'
      endif
```

```
      if(kk.eq.4)then
      open(9,file='uf.dat',status='old')
      read(9,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (9)
      write(*,*)'THE VALUES ENTERED FOR THE U OF F 3-in. SHPB'
      endif
c
C     CURRENT PARAMETERS VALUES ARE SHOWN ON SCREEN FOR THE
C     SYSTEM THAT WAS USED LAST.
c
c
      write(*,*)'                                                '
      write(*,*)'THE PARAMETER DEFINITIONS AND CURRENT VALUES ARE:'
      write(*,81) mm
      write(*,802)
      write(*,82) ni
      write(*,812)
      write(*,83) tstep
      write(*,84) dia
      write(*,805)
      write(*,85) dzi
      write(*,806)
      write(*,86) dzt
      write(*,807)
      write(*,87) velocb
      write(*,88) ro
 81   format('1  Number of terms used in FFT: mm =', i4 )
 802  format('2  Number of points to be analyzed in each pulse:')
 82   format('                              ni =', i4)
 812  format('   The value of ni must not be greater than 1024.')
 83   format('3  Time step:           tstep =', f7.5,1x, 'microsec')
 84   format('4  Diameter of the bar:  dia =', f7.5,1x, 'meters')
 805  format('5  Distance between the gages on the incident bar')
 85   format('   and the specimen:     dzi =', f7.5,1x, 'meters')
 806  format('6  Distance between the gages on the transmitter')
 86   format('   bar and the specimen: dzt =', f7.5,1x, 'meters')
 807  format('7  Longitudinal wave velocity in the bar:')
 87   format('                         velocb =',1x,f7.2,1x,'meters/sec')
 88   format('8  Density of the bar: ro =', f8.2,1x,'Kg/meters cubed')
      write(*,*)'                                                '
c
c
c
 70   write(*,*)'IS THIS THE SHBP SYSTEM YOU WISH TO USE? Y or N '
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n')go to 7
      if(op.ne.'Y'.and.op.ne.'y')go to 70
 76   write(*,*)'Do you wish to change any parameters? Y or N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n') go to 1
      if(op.ne.'Y'.and.op.ne.'y') go to 76
      go to 2
```

```
 7    write(*,*)'                              '
      write(*,*)'THIS IS A DIRECTORY OF THE AVAILABLE SYSTEMS:'
      write(*,*)'                              '
      write(*,*)'1 UNIVERSITY OF FLORIDA 3/4 inch SHPB'
      WRITE(*,*)'2 TYNDALL AIR FORCE BASE'
      WRITE(*,*)'3 NEW SYSTEM OF YOUR CHOICE'
      write(*,*)'4 UNIVERSITY OF FLORIDA 3.0-inch SHPB'
      write(*,*)'                              '
 71   write(*,*)'WHICH SYSTEM DO YOU WISH TO USE?1,2,3,or 4?'
      read(*,*)kk
      if(kk.ne.1.and.kk.ne.2.and.kk.ne.3.and.kk.ne.4)go to 71
      open(12,file='lastsys.dat',status='unknown')
      write(12,150)kk
      close(12)
C
      go to 25
C
C
C     MAKING THE CHANGES...
C     ...OR ENTERING VALUES FOR A NEW SYSTEM (IF 'OTHER' WAS CHOSEN)
C
  2   WRITE(*,*)'                              '
      write(*,*)'   PARAMETER DIRECTORY'
      write(*,*)'                              '
      write(*,*)'1  Number of terms used in Fourier transfer'
      write(*,91) mm
  91  format('   Current value is',6x,i4)
      write(*,*)'2  Number of points to be analyzed.'
      write(*,92) ni
  92  format('   Current value is',6x,i4)
      write(*,*)'3  Time step in microseconds'
      write(*,93) tstep
  93  format('   Current value is',6x,f7.5)
      write(*,*)'4  Diameter of the bar in meters'
      write(*,94) dia
  94  format('   Current value is',6x,f7.5)
      write(*,*)'5  Distance between incident bar gage and the'
      write(*,95) dzi
  95  format('   specimen in meters. Current value is',6x,f7.5)
      write(*,*)'6  Distance between transmitter bar gage and the'
      write(*,95) dzt
      write(*,*)'7  Bar wave velocity in meters/second'
      write(*,97) velocb
  97  format('   Current value is',6x,f7.2)
      write(*,*)'8  Density of the bar in Kg/meters cubed'
      write(*,98) ro
  98  format('   Current value is',5x,f8.2)
      write(*,*)'                              '
 10   write(*,*)'Enter number of parameter to be changed, or 0 if '
      write(*,*)'you are done changing'
      read(*,*)m
      if(m.eq.0) go to 8
```

```
      write(*,*)'Enter new value'
      if(m.eq.1)then
      read(*,*)mm
      endif
      if(m.eq.2)then
      read(*,*)nin
      if(nin.gt.1024) go to 500
      ni=nin
      endif
      if(m.eq.3)then
      read(*,*)tstep
      endif
      if(m.eq.4)then
      read(*,*)dia
      endif
      if(m.eq.5)then
      read(*,*)dzi
      endif
      if(m.eq.6)then
      read(*,*)dzt
      endif
      if(m.eq.7)then
      read(*,*)velocb
      endif
      if(m.eq.8)then
      read(*,*)ro
      endif
      go to 10
C
  500 write(*,*) 'The value of ni that you just entered is greater'
      write(*,*) 'than 1024. It is not permitted.'
      write(*,*) 'Do you wish to quit the job? Y or N'
      read(*,1001) op
      if(op.eq.'Y'.or.op.eq.'y') STOP 'HAVE A NICE TIME'
      write(*,*)'Please check the following parameters again.'
      go to 2
C
  560 STOP 'PLEASE PREPARE INPUT DATA PROPERLY (SEE USERS GUIDE)'
  590 write(*,*)'Do you wish go to the correction phase? Y or N'
      read(*,1001) op
      if(op.eq.'Y'.or.op.eq.'y') go to 502
      if(op.ne.'N'.and.op.ne.'n') go to 590
      STOP 'SORRY, YOU HAVE TO CORRECT ORIGINAL DATA FIRST'
C
C
C     STORING THE NEW VALUES IN THE APPROPRIATE FILES.
C
    8 if(kk.eq.2) then
      open(8,file='tynd.dat',status='unknown')
      write(8,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close(8)
      endif
```

```
      if(kk.eq.4)then
      open(9,file='uf.dat',status='unknown')
      write(9,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (9)
      endif
      if(kk.eq.1)then
      open(10,file='uf2.dat',status='unknown')
      write(10,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (10)
      endif
      if(kk.eq.3)then
      open(11,file='newsys.dat',status='unknown')
      write(11,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close(11)
      endif
c
c
c
C     CALCULATING THE PHASE ANGLE CORRECTIONS PER UNIT
C     DISTANCE FOR THE NEW SYSTEM OR FOR THE MODIFIED OLD SYSTEMS
C
      do 164 i=1,10
      nx=2**i
      if(ni.le.nx)go to 163
 164  continue
 163  nm=nx
      im=i
c
      call angle (tstep,dia,nm,velocb,mm,ph,kk)
      go to 3
C
C     USING OLD PARAMETER VALUES. PHASE ANGLE CORRECTIONS ARE
C     READ FROM FILE ANGLE.DAT,(UF),OR TANGLE.DAT,(TYNDALL).
C
   1  if(kk.eq.2)then
      open(14,file='tangle.dat',status='old')
      do 55 i=1,mm
  55  read(14,17)ph(i)
      close (14)
      endif
      if(kk.eq.1)then
      open(15,file='tuf2.dat',status='old')
      do 9 i=1,mm
   9  read(15,17)ph(i)
      close(15)
      endif
      if(kk.eq.4)then
      open(4,file='angle.dat',status='OLD')
      do 5 i=1,mm
   5  read(4,17)ph(i)
      close(4)
      endif
```

```
c     USER IS ASKED TO INPUT SPECIMENS ID. IT MUST MATCH LETTERS
c     PRECEDING  ...I.DAT,...R.DAT ETC. OF INPUT FILES
c     SPECIMEN ID WILL ALSO BE THE BEGINNING OF THE NAME
c     OF EACH OUTPUT FILE
c
  3   write(*,*)'DO YOU WANT TO SKIP THE CORRECTION PHASE OF'
      write(*,*)'THIS PROGRAM AND GO DIRECTLY TO THE ANALYSIS'
      write(*,*)'PHASE? Y/N'
      read(*,1001)op
      if(op.eq.'Y'.or.op.eq.'y') go to 26
      if(op.ne.'N'.and.op.ne.'n') go to 3
C
 502  write(*,*)'All the input data are ready in units of'
      write(*,*)'microstrain and stored in the files of the'
      write(*,*)'required filenames? Y or N'
      read(*,1001) op
      if(op.eq.'N'.or.op.eq.'n') go to 560
      if(op.ne.'Y'.and.op.ne.'y') go to 502
c
      write(*,*)'Enter number of specimens to be analyzed'
      read(*,*)ks
      do 6 ii=1,ks
      write(*,*)'Enter specimen ID for specimen no.',ii
      read(*,1001)rtitle
c
c
 1001 format(a5)
 17   format(f10.3)
c
c
c
c
c
  4   do 13 k=1,3
c
c     k=1   reading incident pulse data
c     k=2   reading reflected pulse data
c     k=3   reading transmitted pulse data
c
c
      if(k.eq.1)then
      rtitle(4:8)='i.dat'
      dz=dzi
      endif
      if(k.eq.2)then
      rtitle(4:8)='r.dat'
      dz=-dzi
      endif
      if(k.eq.3)then
      rtitle(4:8)='t.dat'
      dz=-dzt
      endif
```

```
      open(k,file=rtitle,status='old')
      do 111 i=1,ni
      read(k,17,end=161)bar(i)
 111  continue
      close(k)
c
 161  do 162 i=ni,nm
      bar(i)=0.
 162  continue
c
      CALL FFT(X,Y,BAR,IM,IM,1)
      DO 12 I=2,55
         XX(I)=SQRT(X(I)*X(I)+Y(I)*Y(I))
         AG(I)=ACOS(X(I)/XX(I))
         IF(Y(I).LT.0)AG(I)=-AG(I)
         AGC(I)=AG(I)+PH(I-1)*dz
         XC(I)=XX(I)*COS(AGC(I))
         YC(I)=XX(I)*SIN(AGC(I))
12    CONTINUE
      DO 33 I=2,55
      X(I)=XC(I)
33    Y(I)=YC(I)
      NM2=NM/2+1
      DO 14 I=NM2,NM
      X(I)=X(NM+1-I)
14    Y(I)=Y(NM+1-I)
      CALL FFT(X,Y,CSP,IM,IM,-1)
c
      if(k.eq.1)rtitle(4:9)='if.dat'
      if(k.eq.2)rtitle(4:9)='rf.dat'
      if(k.eq.3)rtitle(4:9)='tf.dat'
      open(k,file=rtitle,status='unknown')
c
c     WRITE OUTPUT FILES
c
      do 22 i=1,ni
      write(k,17)csp(i)
 22   continue
      close(k)
 13   continue
  6   continue
c
C     THE PHASE CORRECTIONS HAVE BEEN DONE. THE USER HAS NOW THE
C     OPTION TO STOP THE PROGRAM, OR TO CONTINUE  TO CALCULATE
C     STRESSES, STRAINS, ETC...
c
 72   write(*,*)'THE CORRECTIONS ON  ',KS,'SET(S) OF DATA HAVE'
      write(*,*)'BEEN CALCULATED. DO YOU WISH TO PROCEDE WITH'
      write(*,*)'THE STRESS AND STRAIN CALCULATIONS? Y/N'
      read(*,1001)OP
      if(OP.EQ.'N'.or.op.eq.'n')GO TO 66
      if(op.ne.'Y'.and.op.ne.'y')go to 72
```

```
c
 26   ssfac=(velocb**2)*ro*1.e-6
c ssfac is the young's modulus of the bars in MPa
c
 1011 format(a)
 1002 format(f10.3,2(1h,f10.3))
 1003 format(4f10.3)
 1004 format(f6.2,1h,f10.3)
 1005 format(f10.3)
 1006 format(i2)
 1007 format('Enter # ',i2,' specimen ID')
c
c
      AREAB=3.14159*DIA*DIA/4.
      EGCONS=AREAB*VELOCB*TSTEP*SSFAC
c
 75   write(*,*)'Which set of data is to be used'
      write(*,*)'Original(0) or Corrected(1)'
      read(*,*)l
      if(l.ne.0.and.l.ne.1)go to 75
      if(l.eq.0) go to 751
 570  write(*,*)'Is the set of corrected data ready? Y or N'
      read(*,1001) op
      if(op.eq.'N'.or.op.eq.'n') go to 590
      if(op.ne.'Y'.and.op.ne.'y') go to 570
      go to 580
 751  write(*,*)'The original data you are going to use are ready'
      write(*,*)'in units of microstrains and stored in the files'
      write(*,*)'of the required filenames? Y or N'
      read(*,1001) op
      if(op.eq.'N'.or.op.eq.'n') GO TO 560
      if(op.ne.'Y'.and.op.ne.'y') go to 751
 580  write(*,*)'                    '
      write(*,*)'Enter the total number of specimens'
      read(*,*)ns
c
      write(*,*)'ENTER SPECIMEN PARAMETERS FOR EACH SPECIMEN'
      write(*,*)'IN THE FOLLOWING :'
      do 21 i=1,ns
      write(*,1007)i
      read(*,1011)n(i)
      write(*,*)'Enter the specimen length in meters'
      read(*,*)sl(i)
      write(*,*)'Enter the cross section area ratio of the bars to'
      write(*,*)'the specimen'
      read(*,*)ar(i)
  21  continue
c
      do 23 ik=1,ns
      id=n(ik)
      slen=sl(ik)
      aratio=ar(ik)
```

```
c
c     using original set of data (uncorrected)
c
      if(l.eq.0)then
      idi=id//'i.dat'
      idr=id//'r.dat'
      idt=id//'t.dat'
c
c     output files:uncorrected.  (example: if ID is test then
c     output will be testsso.dat)
c
      idsts=id//'sso.dat'
      idstn=id//'sno.dat'
      ideng=id//'eno.dat'
c
c     using corrected set of data
c
      else
      idif=id//'if.dat'
      idrf=id//'rf.dat'
      idtf=id//'tf.dat'
c
c     ouput files
c
      idsts=id//'ssf.dat'
      idstn=id//'snf.dat'
      ideng=id//'enf.dat'
      endif
c
c
c
      if(l.eq.0)open(1,file=idi,status='old')
      if(l.eq.1)open(1,file=idif,status='old')
      do 15 i=1,ni
      read(1,1005)binc(i)
   15 continue
      close(1)
      if(l.eq.0)open(1,file=idr,status='old')
      if(l.eq.1)open(1,file=idrf,status='old')
      do 16 i=1,ni
      read(1,1005)bref(i)
   16 continue
      close(1)
      if(l.eq.0)open(1,file=idt,status='old')
      if(l.eq.1)open(1,file=idtf,status='old')
      do 27 i=1,ni
      read(1,1005)btra(i)
   27 continue
      close(1)
c
   73 write(*,*)'DO YOU WANT TO CALCULATE STRESSES? Y/N'
      read(*,1001)op
```

```
      if(op.eq.'N'.or.op.eq.'n')go to 44
      if(op.ne.'Y'.and.op.ne.'y')go to 73
      open(11,file=idsts,status='unknown')
      do 11 i=1,ni
      strsin=(binc(i)+bref(i))*ssfac*aratio*1.e-6
      strsot=btra(i)*ssfac*aratio*1.e-6
      stress=(strsin+strsot)/2.
      write(11,1002)strsin,strsot,stress
   11 continue
      close(11)
c
      write(*,*)idsts
      write(*,*)'The above is the filename of the output file'
      write(*,*)'containing the specimen stresses at the incident'
      write(*,*)'and transmitter interfaces, and their average.'
      write(*,*)'All are in units of MPa.'
      write(*,*)'                                '
c
   44 write(*,*)'DO YOU WANT TO CALCULATE STRAINS AND VELOCITIES? Y/N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n') go to 23
      if(op.ne.'Y'.and.op.ne.'y') go to 44
      open(22,file=idstn,status='unknown')
      do 112 i=1,ni
      velin=(binc(i)-bref(i))*velocb*1.e-06
      velot=btra(i)*velocb*1.e-06
      strnrt=(velin-velot)/slen
      if(i-1)118,118,19
  118 strain=0.
      go to 20
   19 strain=strain+strnrt*tstep*1.e-04
   20 write(22,1003)strnrt,strain,velin,velot
  112 continue
      close(22)
c
      write(*,*)idstn
      write(*,*)'The above is the filename of the output file'
      write(*,*)'containing the specimen strain rate (1/sec) and'
      write(*,*)'strain (per cent), and the velocities at the'
      write(*,*)'incident and transmitter interfaces (meters/sec).'
      write(*,*)'                                '
c
c
   23 continue

   66 CONTINUE
      STOP 'FINISHED AT LAST.   HAVE A NICE TIME!'
      end
c
C      SUBROUTINE ANGLE
C
C      THIS SUBROUTINE CALCULATES THE PHASE ANGLE CORRECTIONS PER UNIT
```

```
C       DISTANCE, BASED ON THE NUMBER OF POINTS BEING ANALYZED,AND THE
C       BAR WAVE VELOCITY
C
C       PARAMETERS:
C       TSTEP=TIME STEP
C       DIA=DIAMETER OF THE BAR
C       NN=NUMBER OF POINTS USED IN THE ANALYSIS
C       VELOCB=BAR WAVE VELOCITY
C       MM=NUMBER OF TERMS USED IN THE FOURIER ANALYSIS
C       PH=PHASE ANGLE CORRECTIONS RETURNED TO CALLER
C       KK=PARAMETER REFERRING TO SYSTEM BEING USED. NAMELY
C          KK       SYSTEM
c           1       UF 3/4 inch SHPB
c           2       TYNDALL
c           3       OTHER
c           4       UF 3.0-inch SHPB
C
        SUBROUTINE ANGLE(TSTEP,DIA,NN,VELOCB,MM,PH,KK)
        DIMENSION C(100),PH(100)
C
C       FUNCTION STATEMENT; PHASE VELOCITY VS DIA/LAMBDA
C

        CP(V)=.5764+(.4236/(2.148*V**4+.736*V**3-.276*V*V+
     & .3065*V**1.5+1.))
C
        FREQ=1.0/(TSTEP*1.0E-6)
        PI=3.14159
        WO=2.0*PI/NN
C

C       COMPUTE PHASE ANGLE CORRECTIONS
C
   69   W=DIA*WO*FREQ/VELOCB
        V=0.
        I=0
   70   I=I+1
        X=I*W
   80   V=V+0.0001
        XP=2.0*PI*CP(V)*V
   83   IF(XP-X)80,85,85
   85   C(I)=CP(V)
   88   IF(I-MM)70,90,90
   90   DO 100 I=1,MM
        C(I)=C(I)*VELOCB
  100   CONTINUE
C
   17   FORMAT(F10.3)
        if(kk.eq.1) then
        open(15,file='tuf2.dat',status='unknown')
        k=15
        endif
```

```
      if(kk.eq.2)then
      OPEN(14,FILE='TANGLE.DAT',STATUS='UNKNOWN')
      K=14
      endif
      if(kk.eq.3)then
      OPEN(33,FILE='NANGLE.DAT',STATUS='UNKNOWN')
      K=33
      endif
      if(kk.eq.4)then
      OPEN(4,FILE='ANGLE.dat',STATUS='UNKNOWN')
      K=4
      endif
C
      DO 120 I=1,MM
C
C     AK IS THE PHASE ANGLE CORRECTION PER UNIT DISTANCE
C
      AK=I*WO/VELOCB*(VELOCB/C(I)-1.)*FREQ
C
C     STORE ANGLE CORRECTIONS PER UNIT DISTANCE ON FILE ANGLE.DAT,(UF
C     OR ON FILE TANGLE.DAT,(TYNDALL)
C
      PH(I)=AK
      IF(KK.EQ.3) GO TO 110
      WRITE(k,17)PH(I)
 110  CONTINUE
 120  CONTINUE
C
C
      RETURN
      END
C


c.....................................................................
c
c       FAST FOURIER TRANSFORM (FFT) AND INVERSE FFT
c
c.....................................................................
      SUBROUTINE FFT (X,Y,A,M,L,IS)
      DIMENSION X(1),Y(1),A(1)
      L1=L-1
      M1=M-1
      NPTS=2**M1
      FACT1=.5
      FACT2=1./2**M
      IF(IS)2,888,1
888   STOP
1     DO 11 I=1,NPTS
         X(I)=A(2*I-1)
11    Y(I)=-A(2*I)
      CALL FFTP(X,Y,M1,L1)
```

```
      CALL RBITS(X,Y,M1)
      DO 12 I=1,NPTS
         X(I)=X(I)*FACT1
12       Y(I)=-Y(I)*FACT1
      CALL REALTR(X,Y,NPTS,1)
      RETURN
2     CALL REALTR(X,Y,NPTS,-1)
      CALL FFTP(X,Y,M1,L1)
      CALL RBITS(X,Y,M1)
      DO 13 I=1,NPTS
         A(2*I-1)=X(I)*FACT2
13    A(2*I)=Y(I)*FACT2
      RETURN
      END
C.....................................................................
C
C       BIT REVERSAL FOR FFT
C
C.....................................................................
      SUBROUTINE RBITS(X,Y,M)
      DIMENSION X(1),Y(1)
      DIMENSION L(12)
1     DO 70 J=1,12
      IF(J-M)71,71,72
72    L(J)=1
      GO TO 70
71    L(J)=2**(M+1-J)
70    CONTINUE
      L12=L(1)
      L11=L(2)
      L10=L(3)
      L9=L(4)
      L8=L(5)
      L7=L(6)
      L6=L(7)
      L5=L(8)
      L4=L(9)
      L3=L(10)
      L2=L(11)
      L1=L(12)
      JN=1
      DO 60 J1=1,L1
      DO 60 J2=J1,L2,L1
      DO 60 J3=J2,L3,L2
      DO 60 J4=J3,L4,L3
      DO 60 J5=J4,L5,L4
      DO 60 J6=J5,L6,L5
      DO 60 J7=J6,L7,L6
      DO 60 J8=J7,L8,L7
      DO 60 J9=J8,L9,L8
      DO 60 J10=J9,L10,L9
      DO 60 J11=J10,L11,L10
```

```
      DO 60 JR=J11,L12,L11
      IF(JN-JR)61,61,62
61    R=X(JN)
      X(JN)=X(JR)
      X(JR)=R
      F1=Y(JN)
      Y(JN)=Y(JR)
      Y(JR)=F1
62    JN=JN+1
60    CONTINUE
      RETURN
      END
C.....................................................................
C
C       IN PLACE COMPUTATION FOR FFT
C
C.....................................................................
      SUBROUTINE REALTR(A,B,N,ISN)
      DIMENSION A(1),B(1)
      REAL IM
      INC=IABS(ISN)
      NK=N*INC+2
      NH=NK/2
      SD=2.*ATAN(1.)/FLOAT(N)
      CD=2.*SIN(SD)*SIN(SD)
      SD=SIN(SD+SD)
      SN=0.
      IF(ISN.ge.0)GO TO 30
      CN=-1.
      SD=-SD
      GO TO 10
 30   CN=1.
      A(NK-1)=A(1)
      B(NK-1)=B(1)
10    DO 20 J=1,NH,INC
      K=NK-J
      AA=A(J)+A(K)
      AB=A(J)-A(K)
      BB=B(J)-B(K)
      BA=B(J)+B(K)
      RE=CN*BA+SN*AB
      IM=SN*BA-CN*AB
      B(K)=IM-BB
      B(J)=IM+BB
      A(K)=AA-RE
      A(J)=AA+RE
      AA=CN-(CD*CN+SD*SN)
      SN=(SD*CN-CD*SN)+SN
      CN=0.5/(AA*AA+SN*SN)+0.5
      SN=SN*CN
20    CN=CN*AA
      RETURN
```

```
      END
C.....................................................................
C
C       DATA PRUNING BEFORE FFT COMPUTATION
C
C.....................................................................
      SUBROUTINE FFTP(X,Y,M,L)
      DIMENSION X(1),Y(1)
      N=2**M
      L2=2**L
      DO 1 L0=1,M
      LMX=2**(M-L0)
      LMM=LMX
      L1X=2*LMX
      SCL=6.283185/L1X
      IF(L0-M+L)20,30,30
20    LMM=L2
30    DO 1 LM=1,LMM
      ARG=(LM-1)*SCL
      C=COS(ARG)
      S=SIN(ARG)
      DO 1 L1=L1X,N,L1X
      J1=L1-L1X+LM
      J2=J1+LMX
      T1=X(J1)-X(J2)
      T2=Y(J1)-Y(J2)
      X(J1)=X(J2)+X(J1)
      Y(J1)=Y(J1)+Y(J2)
      X(J2)=C*T1+S*T2
1     Y(J2)=C*T2-S*T1
      RETURN
      END
```

```
c.....................................................................
c
c        REVISED MAIN PROGRAM FOR ESL SYSTEM
c.....................................................................
c
c        DATA ACQUISITION, WAVE DISPERSION CORRECTION,
c        DATA ANALYSIS, AND GRAPHICAL RESULTS FOR A SHPB SYSTEM.
c.....................................................................
      DIMENSION   X(1024), Y(1024), XX(56), AG(56)
      DIMENSION  CSP(1024),XC(56),YC(56),AGC(56)
      DIMENSION  BAR(1200),PH(100),ar(100)
      dimension binc(1200),bref(1200),btra(1200),sl(20)
      character*8 idi,idr,idt,name
      character*10 idsts,idstn,ideng
      character*9 idif,idrf,idtf
      character*3 id,n(20)
      CHARACTER*9 RTITLE
      CHARACTER*5 OP
C
C     THE DATA NEEDS TO BE "PREPARED" BEFORE MAIN PART OF
C     SHPB CAN BE RUN :
C
      call prep
C
C     THE PROGRAM TITLE AND MAIN FUNCTION ARE PRINTED ON SCREEN
C
 150  format(i3)
 18   FORMAT(4f7.5,f7.2,2i4,f8.2)
      write(*,*)'THIS IS PROGRAM SHPB. IT PERFORMS THE PHASE '
      write(*,*)'CORRECTIONS FOR DISPERSION ON ANY WAVEFORM'
      write(*,*)'AND DATA ANALYSIS'
      write(*,*)'                                 '
C
C     CHECK TO SEE WHICH SYSTEM WAS USED LAST
C
      open(12,file='sys.dat',status='old')
      read(12,150)kk
      close(12)
 25   write(*,*)'THE PROGRAM PARAMETERS ARE CURRENTLY SET TO    '
C
      if(kk.eq.2)then
      open(8,file='tynd.dat',status='old')
      read(8,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (8)
      WRITE(*,*)'THE VALUES ENTERED FOR TYNDALL 2-inch SHPB'
      endif
C
      if(kk.eq.1)then
      open(10,file='uf2.dat',status='old')
      read(10,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (10)
      write(*,*)'THE VALUES ENTERED FOR THE U OF F 3/4 in. SHPB'
      endif
C
      if(kk.eq.3)go to 2
```

```
c
c
c
      if(kk.eq.4)then
      open(9,file='UF.dat',status='old')
      read(9,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (9)
      write(*,*)'THE VALUES ENTERED FOR THE U OF F 3-in. SHPB'
      endif
c
C     CURRENT PARAMETERS VALUES ARE SHOWN ON SCREEN FOR THE
C     SYSTEM THAT WAS USED LAST.
c
c
      write(*,*)'PARAMETER DEFINITIONS AND CURRENT VALUES ARE:'
      write(*,*)'tstep=time step =', tstep,' microsec'
      write(*,*)'ni=number of points to be analyzed in each pulse.'
      write(*,*)'it is given by the base period divided by the time'
      write(*,*)'step. It MUST be a power of 2. ni=',ni
      write(*,*)'dia=diameter of bar=',dia,' meters'
      write(*,*)'dzi=distance between strain gage on the incident '
      write(*,*)'    bar and the specimen.dzi=',dzi,' meters'
      write(*,*)'dzt=distance between strain gage on the transmitter'
      write(*,*)'    bar and the specimen.dzt=',dzt,' meters'
      write(*,*)'velocb=longitudinal wave velocity in the bar'
      write(*,*)'velocb=',velocb,' meters/sec'
      write(*,*)'mm=number of frequencies in Fourier analysis=',mm
      write(*,*)'ro=density of the bar. ro=',ro,' Kg/meters cubed'
c
      write(*,*)'                              '
c
      write(*,*)'IS THIS THE SHBP SYSTEM YOU WISH TO USE? Y or N '
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n')go to 7
c
c
      write(*,*)'Do you wish to change any parameters? Y or N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n') go to 1
      go to 2
 7    write(*,*)'                              '
      write(*,*)'THIS IS A DIRECTORY OF THE AVAILABLE SYSTEMS:'
      write(*,*)'                              '
      write(*,*)'1 UNIVERSITY OF FLORIDA 3/4 inch SHPB'
      WRITE(*,*)'2 TYNDALL AIR FORCE BASE'
      WRITE(*,*)'3 NEW SYSTEM OF YOUR CHOICE'
      write(*,*)'4 UNIVERSITY OF FLORIDA 3.0-inch SHPB'
      write(*,*)'                              '
      write(*,*)'WHICH SYSTEM DO YOU WISH TO USE?1,2,3,or 4?'
      read(*,*)kk
      open(12,file='sys.dat',status='unknown')
      write(12,150)kk
      close(12)
C
      go to 25
```

```
C
C
C
C     MAKING THE CHANGES...
C     ...OR ENTERING VALUES FOR A NEW SYSTEM (IF 'OTHER' WAS CHOSEN)
C
  2   WRITE(*,*)'                              '
      write(*,*)'   PARAMETER DIRECTORY'
      write(*,*)'                              '
      write(*,*)'1  Number of terms used in Fourier transfer'
      write(*,*)'   current value is',mm
      write(*,*)'2  Number of points to be analyzed.'
      write(*,*)'   It MUST be a power of 2. Current value is',ni
      write(*,*)'3  Time step in microseconds'
      write(*,*)'   current value is',tstep
      write(*,*)'4  Diameter of the bar in meters'
      write(*,*)'   current value is',dia
      write(*,*)'5  Distance between incident bar gage and the'
      write(*,*)'   specimen, in meters. Current value is',dzi
      write(*,*)'6  Distance between transmitter bar gage and the'
      write(*,*)'   specimen, in meters. Current value is',dzt
      write(*,*)'7  Bar wave velocity in meters/second'
      write(*,*)'   current value is',velocb
      write(*,*)'8  Density of the bar in Kg/meters cubed'
      write(*,*)'   current value is',ro
      write(*,*)'                              '
 10   write(*,*)'Enter number of parameter to be changed, or 0 if '
      write(*,*)'you are done changing'
      read(*,*)m
      if(m.eq.0) go to 8
      write(*,*)'Enter new value'
      if(m.eq.1)then
      read(*,*)mm
      endif
      if(m.eq.2)then
      read(*,*)ni
      endif
      if(m.eq.3)then
      read(*,*)tstep
      endif
      if(m.eq.4)then
      read(*,*)dia
      endif
      if(m.eq.5)then
      read(*,*)dzi
      endif
      if(m.eq.6)then
      read(*,*)dzt
      endif
      if(m.eq.7)then
      read(*,*)velocb
      endif
      if(m.eq.8)then
      read(*,*)ro
      endif
```

```
      go to 10
C
C
C     STORING THE NEW VALUES IN THE APPROPRIATE FILES.
C
  8   if(kk.eq.2) then
      open(8,file='tynd.dat',status='unknown')
      write(8,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close(8)
      endif
      if(kk.eq.4)then
      open(9,file='UF.dat',status='unknown')
      write(9,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (9)
      endif
      if(kk.eq.1)then
      open(10,file='uf2.dat',status='unknown')
      write(10,18)tstep,dia,dzi,dzt,velocb,mm,ni,ro
      close (10)
      endif
C
C
C
C     CALCULATING THE PHASE ANGLE CORRECTIONS PER UNIT
C     DISTANCE FOR THE NEW SYSTEM OR FOR THE MODIFIED OLD SYSTEMS
C
      call angle (tstep,dia,ni,velocb,mm,ph,kk)
      go to 3
C
C     USING OLD PARAMETER VALUES. PHASE ANGLE CORRECTIONS ARE
C     READ FROM FILE ANGLE.DAT,(UF),OR TANGLE.DAT,(TYNDALL).
C
  1   if(kk.eq.2)then
      open(14,file='tangle.dat',status='old')
      do 55 i=1,mm
 55   read(14,17)ph(i)
      close (14)
      endif
      if(kk.eq.1)then
      open(15,file='tuf2.dat',status='old')
      do 9 i=1,mm
  9   read(15,17)ph(i)
      close(15)
      endif
      if(kk.eq.4)then
      open(4,file='angle.dat',status='OLD')
      do 5 i=1,mm
  5   read(4,17)ph(i)
      close(4)
      endif
C
C
C     USER IS ASKED TO INPUT SPECIMENS ID. IT MUST MATCH LETTERS
C     PRECEDING  ...I.DAT,...R.DAT ETC. OF INPUT FILES
C     SPECIMEN ID WILL ALSO BE THE BEGINNING OF THE NAME
```

```
c      OF EACH OUTPUT FILE
c
c
  3    write(*,*)'DO YOU WANT TO SKIP THE CORRECTION PHASE OF'
       write(*,*)'OF THIS PROGRAM AND GO DIRECTLY TO THE ANALYSIS'
       write(*,*)'PHASE? Y/N'
       read(*,1001)op
       if(op.eq.'Y'.or.op.eq.'y') go to 26
       if(op.ne.'N'.and.op.ne.'n') go to 3
c
       write(*,*)'Enter number of specimens to be analyzed'
       read(*,*)ks
       do 6 ii=1,ks
       write(*,*)'Enter specimen ID for specimen no.',ii
       read(*,1001)rtitle
c
c
 1001  format(a5)
 17    format(f10.3)
c
c
  4    do 13 k=1,3
c
c      k=1   reading incident pulse data
c      k=2   reading reflected pulse data
c      k=3   reading transmitted pulse data
c
       if(k.eq.1)then
       rtitle(4:8)='i.dat'
       dz=dzi
       endif
       if(k.eq.2)then
       rtitle(4:8)='r.dat'
       dz=-dzi
       endif
       if(k.eq.3)then
       rtitle(4:8)='t.dat'
       dz=-dzt
       endif
       open(k,file=rtitle,status='old')
       do 111 i=1,ni
       read(k,17)bar(i)
 111   continue
       close(k)
       CALL FFT(X,Y,BAR,10,10,1)
       DO 12 I=2,55
          XX(I)=SQRT(X(I)*X(I)+Y(I)*Y(I))
          AG(I)=ACOS(X(I)/XX(I))
          IF(Y(I).LT.0)AG(I)=-AG(I)
          AGC(I)=AG(I)+PH(I-1)*dz
          XC(I)=XX(I)*COS(AGC(I))
          YC(I)=XX(I)*SIN(AGC(I))
12     CONTINUE
       DO 33 I=2,55
       X(I)=XC(I)
```

```
33      Y(I)=YC(I)
        DO 14 I=513,1024
        X(I)=X(1025-I)
14      Y(I)=Y(1025-I)
        CALL FFT(X,Y,CSP,10,10,-1)
c
        if(k.eq.1)rtitle(4:9)='if.dat'
        if(k.eq.2)rtitle(4:9)='rf.dat'
        if(k.eq.3)rtitle(4:9)='tf.dat'
        open(k,file=rtitle,status='unknown')
c
c       WRITE OUTPUT FILES
c
        do 22 i=1,ni
        write(k,17)csp(i)
 22     continue
        close(k)
 13     continue
  6     continue
c
C       THE PHASE CORRECTIONS HAVE BEEN DONE. THE USER HAS NOW THE
C       OPTION TO STOP THE PROGRAM, OR TO CONTINUE  TO CALCULATE
C       STRESSES, STRAINS, ETC...
c
 72     write(*,*)'THE CORRECTIONS ON',ks ,' SET(S) OF DATA HAVE'
        write(*,*)'BEEN CALCULATED. DO YOU WISH TO PROCEDE WITH'
        write(*,*)'THE STRESS AND STRAIN CALCULATIONS? Y/N'
        read(*,1001)OP
        if(OP.EQ.'N'.or.op.eq.'n')GO TO 66
        if(op.ne.'Y'.and.op.ne.'y') go to 72
c
 26     ssfac=(velocb**2)*ro*1.0e-6
c
 1011 format(a)
 1002 format(3f10.3)
 1003 format(4f10.3)
c1004 format(f6.2,1h,f10.3)
 1005 format(f10.3)
 1006 format(i2)
 1007 format(' enter # ',i2,' specimen ID')
c
        AREAB=3.14159*DIA*DIA/4.
        EGCONS=AREAB*VELOCB*TSTEP*SSFAC
c
  75    write(*,*)'Which set of data is to be used'
        write(*,*)'Original(0) or Corrected(1)'
        read(*,*)l
        if(l.ne.0.and.l.ne.1)GO TO 75
        write(*,*)'Enter the total number of specimens'
        read(*,*)ns
        do 21 i=1,ns
        write(*,1007)i
        read(*,1011)n(i)
        write(*,*)'specimen length, in meters'
        read(*,*)sl(i)
```

```
      write(*,*)'Enter cross section area ratio of bars to the specimen'
      read(*,*)ar(i)
   21 continue
c
      do 23 ik=1,ns
      id=n(ik)
      slen=sl(ik)
      aratio=ar(ik)
c
c     using original set of data (uncorrected)
c
      if(l.eq.0)then
      idi=id//'i.dat'
      idr=id//'r.dat'
      idt=id//'t.dat'
c
c     output files:uncorrected.  (example: if ID is test then
c     output will be testsso.dat)
c
      idsts=id//'sso.dat'
      idstn=id//'sno.dat'
      ideng=id//'eno.dat'
c
c     using corrected set of data
c
      else
      idif=id//'if.dat'
      idrf=id//'rf.dat'
      idtf=id//'tf.dat'
c
c     ouput files
c
      idsts=id//'ssf.dat'
      idstn=id//'snf.dat'
      ideng=id//'enf.dat'
      endif
c
c
c
      if(l.eq.0)open(1,file=idi,status='old')
      if(l.eq.1)open(1,file=idif,status='old')
      do 15 i=1,ni
      read(1,1005)binc(i)
   15 continue
      close(1)
      if(l.eq.0)open(1,file=idr,status='old')
      if(l.eq.1)open(1,file=idrf,status='old')
      do 16 i=1,ni
      read(1,1005)bref(i)
   16 continue
      close(1)
      if(l.eq.0)open(1,file=idt,status='old')
      if(l.eq.1)open(1,file=idtf,status='old')
      do 27 i=1,ni
      read(1,1005)btra(i)
```

```
  27  continue
      close(1)
c
  73  write(*,*)'DO YOU WANT TO CALCULATE STRESSES? Y/N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n')go to 44
      if(op.ne.'Y'.and.op.ne.'y')go to 73
      open(11,file=idsts,status='unknown')
      do 11 i=1,ni
      strsin=(binc(i)+bref(i))*ssfac*aratio*1.e-6
      strsot=btra(i)*ssfac*aratio*1.e-6
      stress=(strsin+strsot)/2.
      write(11,1002)strsin,strsot,stress
  11  continue
      close(11)
c
      write(*,*)idsts
c
  44  write(*,*)'DO YOU WANT TO CALCULATE STRAINS AND VELOCITIES? Y/N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n') go to 23
      if(op.ne.'Y'.and.op.ne.'y')go to 44
      open(22,file=idstn,status='unknown')
      do 112 i=1,ni
      velin=(binc(i)-bref(i))*velocb*1.e-06
      velot=btra(i)*velocb*1.e-06
      strnrt=(velin-velot)/slen
      if(i-1)118,118,19
 118  strain=0.
      go to 20
  19  strain=strain+strnrt*tstep*1.e-04
  20  write(22,1003)strnrt,strain,velin,velot
 112  continue
      close(22)
c
      write(*,*)idstn
c
c
  23  continue

  66  CONTINUE
      write(*,*)'FINISHED WITH STRESS AND STRAIN CALCULATIONS'
      write(*,*)'At this point you can either exit this '
      write(*,*)'program and use program SEC to prepare the'
      write(*,*)'plots you desire to see. Or you can stay with'
      write(*,*)'this program which will set up all possible '
      write(*,*)'plots. Time step will be 0.5e-6 sec.'
      write(*,*)'EXIT? Y/N'
      read(*,1001)op
      if(op.eq.'N'.or.op.eq.'n') then
      call sec
      go to 67
      endif
      if(op.ne.'Y'.and.op.ne.'y')go to 66
  67  continue
```

```
      end
c
c
c     ..................................................................
c
c     This subroutine reads the data from SHPB output and sets
c     up nine files in a format such that VUPOINT and LOTUS
c     can recognize and plot.  The nine files are : stress1 and
c     stress2 vs. time, avg. stress vs. time, strain vs. time,
c     strain rate vs. time, velocity1 and velocity2 vs. time,
c     stress1 and stress2 vs. strain.
c
      subroutine sec
c
      dimension rd(1200),time(0:1200),s1(1200),s2(1200)
      dimension strain(1200),str(1200),vel1(1200),vel2(1200)
      character*5 op
      character*10 idsts,idstn
      character*3  id
      character*9 r2,r3,r4,r7,r8,r9,r10,r11,r20
c
      write(*,*)'enter specimen id '
      read(*,1001)id
c
      idsts=id//'ssf.dat'
      r2='st1t.ad'
c
      idstn=id//'snf.dat'
      r3='st2t.ad'
c
      r8='avst.ad'
c
      r9='strat.ad'
c
      r4='stnt.ad'
c
      r10='vel1.ad'
c
      r11='vel2.ad'
c
      r20='ststt2.ad'
c
      r7='ststt.ad'
c
      open(12,file=idstn,status='old')
      open(1,file=idsts,status='old')
      open(2,file=r2,status='unknown')
      open(3,file=r3,status='unknown')
      open(4,file=r4,status='unknown')
c
c
      time(1)=0.0
      do 2 i=1,1024
c
      read(1,1002)s1(i),s2(i),rd(i)
```

```
c
      read(12,1003)str(i),strain(i),vel1(i),vel2(i)
c
c
c
c
      ii=i-1
      IF(ii.eq.0)go to 5
      time(i)=time(ii)+0.5e-6
    5 write(2,100) time(i),s1(i)
      write(3,100) time(i),s2(i)
      write(4,100) time(i),strain(i)
    2 continue
      close (1)
      close (12)
      close (2)
      close (3)
      close (4)
      open(7,file=r7,status='unknown')
      open(8,file=r8,status='unknown')
      open(9,file=r9,status='unknown')
      open(12,file=idstn,status='old')
      open(1,file=idsts,status='old')
      time(1)=0.0
      do 3 i=1,1024
c
      read(1,1002)s1(i),s2(i),rd(i)
c
      read(12,1003)str(i),strain(i),vel1(i),vel2(i)
c
c
c
c
      ii=i-1
      IF(ii.eq.0)go to 8
      time(i)=time(ii)+0.5e-6
    8 write(8,100) time(i),rd(i)
      write(9,100) time(i),str(i)
      write(7,100) strain(i),s1(i)
    3 continue
      close (1)
      close (8)
      close (9)
      close (7)
      close (12)
      open(10,file=r10,status='unknown')
      open(20,file=r20,status='unknown')
      open(11,file=r11,status='unknown')
      open(12,file=idstn,status='old')
      open(1,file=idsts,status='old')
      time(1)=0.0
      do 12 i=1,1024
c
      read(1,1002)s1(i),s2(i),rd(i)
c
```

```
      read(12,1003)str(i),strain(i),vel1(i),vel2(i)
c
c
c
c
      ii=i-1
      IF(ii.eq.0)go to 18
      time(i)=time(ii)+0.5e-6
  18    write(10,100) time(i),vel1(i)
      write(11,100) time(i),vel2(i)
      write(20,100) strain(i),s2(i)
  12  continue
 100  format(e10.3,1x,e10.3)
   17 format(f10.3)
c
 1001 format(a)
 1002 format(3f10.3)
 1003 format(4f10.3)
      close (1)
      close (10)
      close (20)
      close (12)
      close (11)
c
      return
      end
c
c     ..................................................................
..
c
C     This subroutine reads the files as recorded by VUPOINT,
c     converts data to the right units, and sets up input
c     files that SHPB can read.
c
       subroutine prep

       dimension rdata(1200),rd(1200)
       character*9 rtitle,sample
       character*5 op
c
       write(*,*)'enter specimen id'
       read(*,1001)rtitle
  10   write(*,*)' is this an incident pulse? Y/N'
       read(*,1001)op
       if(op.eq.'Y'.or.op.eq.'y')then
       k=1
       rtitle(4:8)='i.dat'
       sample='inc.ad'
       go to 5
       endif
       write(*,*)' is this a reflected pulse? Y/N'
       read(*,1001)op
       if(op.eq.'Y'.or.op.eq.'y')then
       k=2
       rtitle(4:8)='r.dat'
       sample='refl.ad'
```

```
      go to 5
      endif
      write(*,*)' is this a transmitted pulse? Y/N'
      read(*,1001)op
      if(cp.eq.'N'.or.op.eq.'n')go to 10
      if(op.eq.'Y'.or.op.eq.'y')then
      k=3
      rtitle(4:8)='t.dat'
      sample='trans.ad'
      endif
c
    5 write(*,*)' enter calibration factor, microstrains/volts'
      write(*,*)' DONT FORGET TO INCLUDE MAG. FACTOR!!!'
      read(*,*)cal
c
c     READING DATA FROM VUPOINT FILES...
c
      open(8,file=sample,status='old')
      read(8,100)rd(1)
  100 format(////////////12x,e11.4)
  200 format(12x,e11.4)
 1001 format(a5)
   17 format(f10.3)
      do 1 i=2,1024
      read(8,200,end=2)rdata(i)
c
c     MULTIPLYING BY THE CALIBRATION FACTOR...
c
      rd(i)=rdata(i)*cal
    1 continue
    2 do 3 n=i,1024
    3 rd(n)=0.0
      close (2)
      close (8)
c
      open(k,file=rtitle,status='unknown')
      alfa=1.0
      write(*,*)'is this a tensile test? if the incident'
      write(*,*)'pulse is NOT positive you'
      write(*,*)' need to invert the waveform(s). Invert? Y/N'
      read(*,1001)op
      if(op.eq.'Y'.or.op.eq.'y')alfa=-1.0
c
c     INVERTING THE WAVEFORM...
c
      do 6 i=1,1024
      rd(i)=rd(i)*alfa
    6 write(k,17)rd(i)
      close (k)
c
c     LOOP THROUGH THIS SUBROUTINE UNTIL ALL
c     THREE WAVEFORMS HAVE BEEN PROCESSED
c
      write(*,*)' Have you prepared all three pulses? Y/N'
      read(*,1001)op
```

```
      if(op.eq.'N'.or.op.eq.'n')go to 10
      return
      end
C     ..................................................................
C
C     SUBROUTINE ANGLE
C
C     THIS SUBROUTINE CALCULATES THE PHASE ANGLE CORRECTIONS PER UNIT
C     DISTANCE, BASED ON THE NUMBER OF POINTS BEING ANALYZED,AND THE
C     BAR WAVE VELOCITY
C
C     PARAMETERS:
C     TSTEP=TIME STEP
C     DIA=DIAMETER OF THE BAR
C     NN=NUMBER OF POINTS USED IN THE ANALYSIS
C     VELOCB=BAR WAVE VELOCITY
C     MM=NUMBER OF TERMS USED IN THE FOURIER ANALYSIS
C     PH=PHASE ANGLE CORRECTIONS RETURNED TO CALLER
C     KK=PARAMETER REFERRING TO SYSTEM BEING USED. NAMELY
C        KK       SYSTEM
c         1       UF 3/4 inch SHPB
c         2       TYNDALL
c         3       OTHER
c         4       UF 3.0-inch SHPB
C
      SUBROUTINE ANGLE(TSTEP,DIA,NN,VELOCB,MM,PH,KK)
      DIMENSION C(100),PH(100)
C
C     FUNCTION STATEMENT; PHASE VELOCITY VS DIA/LAMBDA
C

      CP(V)=.5764+(.4236/(2.148*V**4+.736*V**3-.276*V*V+
     & .3065*V**1.5+1.))
C
      FREQ=1.0/(TSTEP*1.0E-6)
      PI=3.14159
      WO=2.0*PI/NN
C

C     COMPUTE PHASE ANGLE CORRECTIONS
C
   69 W=(DIA/2.)*WO*FREQ/VELOCB
      V=0.
      I=0
   70 I=I+1
      X=I*W
   80 V=V+0.0001
      XP=2.0*PI*CP(V)*V
   83 IF(XP-X)80,85,85
   85 C(I)=CP(V)
   88 IF(I-MM)70,90,90
   90 DO 100 I=1,MM
      C(I)=C(I)*VELOCB
  100 CONTINUE
C
```

```
 17     FORMAT(F10.3)
        if(kk.eq.1) then
        open(15,file='tuf2.dat',status='unknown')
        k=15
        endif
        if(kk.eq.2)then
        OPEN(14,FILE='TANGLE.DAT',STATUS='UNKNOWN')
        K=14
        endif
        if(kk.eq.4)then
        OPEN(4,FILE='ANGLE.dat',STATUS='UNKNOWN')
        K=4
        endif
C
        DO 120 I=1,MM
C
C       AK IS THE PHASE ANGLE CORRECTION PER UNIT DISTANCE
C
        AK=I*WO/VELOCB*(VELOCB/C(I)-1.)*FREQ
C
C       STORE ANGLE CORRECTIONS PER UNIT DISTANCE ON FILE ANGLE.DAT,(UF)
C       OR ON FILE TANGLE.DAT,(TYNDALL)
C
        PH(I)=AK
        IF(KK.EQ.3) GO TO 110
        WRITE(k,17)PH(I)
 110    CONTINUE
 120    CONTINUE
c
C
        RETURN
        END
C


c.....................................................................
c
c         FAST FOURIER TRANSFORM (FFT) AND INVERSE FFT
c
c.....................................................................
       SUBROUTINE FFT (X,Y,A,M,L,IS)
       DIMENSION X(1),Y(1),A(1)
       L1=L-1
       M1=M-1
       NPTS=2**M1
       FACT1=.5
       FACT2=1./2**M
       IF(IS)2,888,1
888    STOP
1      DO 11 I=1,NPTS
          X(I)=A(2*I-1)
11     Y(I)=-A(2*I)
       CALL FFTP(X,Y,M1,L1)
       CALL RBITS(X,Y,M1)
       DO 12 I=1,NPTS
```

```
         X(I)=X(I)*FACT1
12       Y(I)=-Y(I)*FACT1
      CALL REALTR(X,Y,NPTS,1)
      RETURN
2     CALL REALTR(X,Y,NPTS,-1)
      CALL FFTP(X,Y,M1,L1)
      CALL RBITS(X,Y,M1)
      DO 13 I=1,NPTS
         A(2*I-1)=X(I)*FACT2
13    A(2*I)=Y(I)*FACT2
      RETURN
      END
C.....................................................................
C
C       BIT REVERSAL FOR FFT
C
C.....................................................................
      SUBROUTINE RBITS(X,Y,M)
      DIMENSION X(1),Y(1)
      DIMENSION L(12)
1     DO 70 J=1,12
      IF(J-M)71,71,72
72    L(J)=1
      GO TO 70
71    L(J)=2**(M+1-J)
70    CONTINUE
      L12=L(1)
      L11=L(2)
      L10=L(3)
      L9=L(4)
      L8=L(5)
      L7=L(6)
      L6=L(7)
      L5=L(8)
      L4=L(9)
      L3=L(10)
      L2=L(11)
      L1=L(12)
      JN=1
      DO 60 J1=1,L1
      DO 60 J2=J1,L2,L1
      DO 60 J3=J2,L3,L2
      DO 60 J4=J3,L4,L3
      DO 60 J5=J4,L5,L4
      DO 60 J6=J5,L6,L5
      DO 60 J7=J6,L7,L6
      DO 60 J8=J7,L8,L7
      DO 60 J9=J8,L9,L8
      DO 60 J10=J9,L10,L9
      DO 60 J11=J10,L11,L10
      DO 60 JR=J11,L12,L11
      IF(JN-JR)61,61,62
61    R=X(JN)
      X(JN)=X(JR)
      X(JR)=R
```

```
      F1=Y(JN)
      Y(JN)=Y(JR)
      Y(JR)=F1
62    JN=JN+1
60    CONTINUE
      RETURN
      END
C..... .....
C
C       IN PLACE COMPUTATION FOR FFT
C
C.....
      SUBROUTINE REALTR(A,B,N,ISN)
      DIMENSION A(1),B(1)
      REAL IM
      INC=IABS(ISN)
      NK=N*INC+2
      NH=NK/2
      SD=2.*ATAN(1.)/FLOAT(N)
      CD=2.*SIN(SD)*SIN(SD)
      SD=SIN(SD+SD)
      SN=0.
      IF(ISN.ge.0)GO TO 30
      CN=-1.
      SD=-SD
      GO TO 10
 30   CN=1.
      A(NK-1)=A(1)
      B(NK-1)=B(1)
10    DO 20 J=1,NH,INC
      K=NK-J
      AA=A(J)+A(K)
      AB=A(J)-A(K)
      BB=B(J)-B(K)
      BA=B(J)+B(K)
      RE=CN*BA+SN*AB
      IM=SN*BA-CN*AB
      B(K)=IM-BB
      B(J)=IM+BB
      A(K)=AA-RE
      A(J)=AA+RE
      AA=CN-(CD*CN+SD*SN)
      SN=(SD*CN-CD*SN)+SN
      CN=0.5/(AA*AA+SN*SN)+0.5
      SN=SN*CN
20    CN=CN*AA
      RETURN
      END
```

```
C.....................................................................
C
C       DATA PRUNING BEFORE FFT COMPUTATION
C
C.....................................................................
      SUBROUTINE FFTP(X,Y,M,L)
      DIMENSION X(1),Y(1)
      N=2**M
      L2=2**L
      DO 1 L0=1,M
      LMX=2**(M-L0)
      LMM=LMX
      L1X=2*LMX
      SCL=6.283185/L1X
      IF(L0-M+L)20,30,30
20    LMM=L2
30    DO 1 LM=1,LMM
      ARG=(LM-1)*SCL
      C=COS(ARG)
      S=SIN(ARG)
      DO 1 L1=L1X,N,L1X
      J1=L1-L1X+LM
      J2=J1+LMX
      T1=X(J1)-X(J2)
      T2=Y(J1)-Y(J2)
      X(J1)=X(J2)+X(J1)
      Y(J1)=Y(J1)+Y(J2)
      X(J2)=C*T1+S*T2
1     Y(J2)=C*T2-S*T1
      RETURN
      END
```

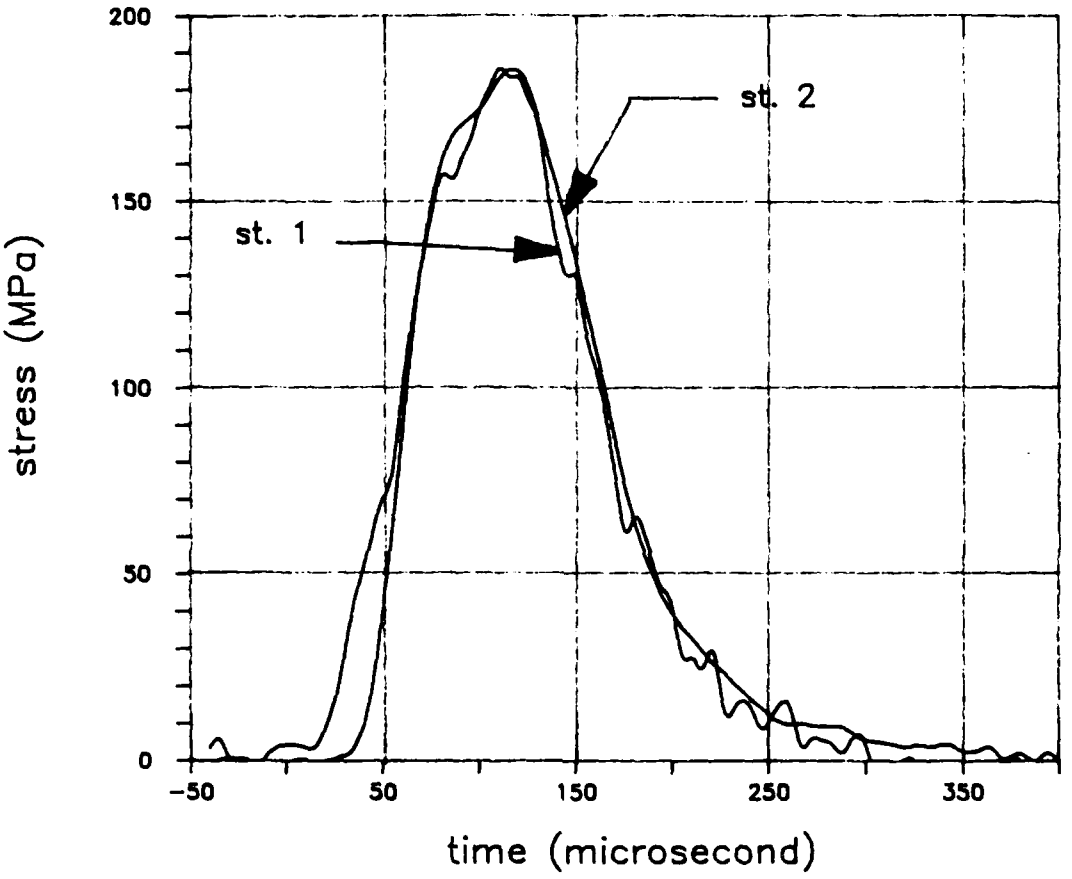


Figure 1. Demonstration of graphics output with Freelance & Lotus 1-2-3